★CACM 中国版★

计算机协会通讯

CACM.ACM.ORG

2015 年 4 月第 58 卷第 4 期

Sketch-Thru-Plan
指挥和控制使用的多通道界面

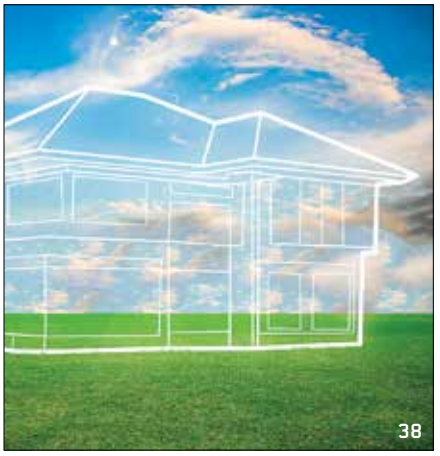
谁会在没有蓝图的时候修建房屋?

医疗设备安全

选择静态卷积引擎

Association for Computing Machinery

观点



38 **观点**
谁会在没有蓝图的时候修建房屋?
通过考虑问题及其解决方案来找出更好的解决方案，而不是仅仅考虑代码。
Leslie Lamport

实践

42 **静态与失败之间的抉择**
最终发现，动态系统的安全性仅仅是低了一点。
Paul Vixie

投稿文章

56 **Sketch-Thru-Plan 指挥和控制使用的多通道界面**
使用军事术语后，用户可以创建标签和绘制符号，从而把对象定位在数字化的地图上。
Philip R. Cohen，Edward C. Kaiser，M. Cecelia Buchanan，Scott Lind，Michael J. Corrigan，R. Matthews Wesson

在本《通讯（Communications）独家视频中，您可以观看作者对本研究的讨论。

评论文章

74 **医疗设备的安全性挑战**
植入性设备常常依赖软件，它们挽救了无数的生命。然而它们有多安全?
Johannes Sametinger、Jerzy Rozenblit、Roman Lysecky 和 Peter Ott

研究亮点

84 **技术视角**
计算机硬件的专用化趋势
Trevor Mudge

85 **卷积引擎：平衡专用计算的效率与灵活性**
Wajahat Qadeer、Rehan Hameed、Ofer Shacham、Preethi Venkatesan、Christos Kozyrakis 和 Mark Horowitz

关于封面：
本月的封面故事介绍了一个名为 Sketch-Thru-Plan 的高级多模式系统。该系统目前用于快速制定地面军事行动计划。STP 将基于地图的界面和移动界面与替代输入模式，如语音、触摸、书写等等相融合。封面插图由 Justin Metz 提供。

从左至右：图片由 SAKONBOON SANSRI、MICHAEL GLENWOOD 和 MOPTIC 提供

ACM 通讯

精选文章中文译文

ACM中国理事会办公室

中国北京清华大学
东主楼 11-236 室
邮编：100084
电话：+86-10-62785025
电子邮件：acmchina@acm.org
联系人：辛爽

ACM通讯

(ISSN 0001-0782) 由计算机协会（2 Penn Plaza, Suite 701, New York, NY 10121-0701）按月发行。

Association for Computing Machinery

DOI:10.1145/2736348

Leslie Lamport

观点：谁会在没有蓝图的时候修建房屋?

通过考虑问题及其解决方案来找出更好的解决方案，而不是仅仅考虑代码。

我从 **1957 年开始**编写程序。但在过去的四十年里，我一直从事计算机科学的研究工作，只写过一点点代码。我创建了 TLA + 规约语言。本文中的论述基于我在编程和帮助工程师编写规约说明时积累的经验。这些经验都是老生常谈；但是我们需要重复古老的箴言，不然愚蠢的思想会占据所有的注意力。我并不编写安全攸关的程序，所以我觉得编写那些程序的人从本文中学不到什么东西。

在砌上一块砖或钉上一颗钉子之前，建筑师要绘制详细的计划。但是，在开始编码前，很少有程序员哪怕是用笔粗略地勾画出程序将要做什么。我们可以向建筑师学习。

程序的蓝图称为*规约说明*。用建筑师的蓝图来比喻软件规约说明书相当有用。例如，它揭示了下列论点中的谬误：因为人们无法用规约说明生成代码，所以规约说明没用。即使无法用蓝图自动构建建筑物，建筑师仍然觉得蓝图有用。不过，比喻可能会让人误解，我并不是说，仅仅因为建筑师绘制蓝图，所以我们就应该编写规约说明。

规约说明书的需求源自两方面的情况。首先，在做事前考虑我们要做什么是一个好想法，正如漫画家 Guindon 所写的那样：“写作是人了解自身想法有多马虎的自然之道。”

为了理解我们正在着手的工作，我们要思考。如果我们理解了事物，我们就能用笔把它解释清楚。如果我们没有用笔解释，那我们就无法知道我们是否真正理解。

第二种情况是，要想写好程序，我们需要跳出代码层面来思考。程序员花费很多时间思考如何编码，

图片由 SAKONBOON SA-NSRI 提供

人们也提出了很多编码方法：如测试驱动的开发、敏捷编程等等。但是，如果程序员只知道冒泡排序这一种排序算法，上面的方法都无法让程序员得出时间为 $O(n \log n)$ 的排序代码。它也无法把某个程序应该如何工作的极度复杂概念转变成简单的、易于维护的代码。在开始编码前，我们需要从更高的层级理解我们的编程任务。

规约说明通常意味着用形式语言书写的材料，它具有精确的句法和（希望是）精确的语义。但是，形式化的规约说明只是全局中的一个方面。建筑师不会为工具房和大桥绘制相同类型的蓝图。我推测，程序员编写的 95% 的代码都足够简单，用两三句散文化的句子足以描绘清楚。另一方面，分布式系统的复杂程度可能与大桥不相上下。它可能需要很多规约说明，某些是形式化的；建筑大桥的蓝图也不止一张。多线程和分布式系统容易出错，所以需要用形式化的规约说明来避免其中的同步错误。（见本期第 66 页 Newcombe 等人的文章）

编写形式化的规约说明的主要原因是要应用工具对它进行检查。工具无法找出在非形式化的规约说明中存在的设计错误，即便您不需要编写形式化的规约说明，您也应该学习编写的方法。当您真正需要编写时，您将没有时间学习编写的方法。在过去的十二年里，我为我的代码写了约六次形式化的规约说明。例如，有一次我必须编写代码来计算图的连通分量。我发现了一个标准算法，但是需要稍作修改，以满足我的需要。这些修改看起来相当简单，但是我决定用 TLA+ 详细说明和检查修改后的算法。我用了一整天的时间才让算法正确无误。与使用常规的程序调试工具调试 Java 实现相比，用诸如 TLA+ 的更高级语言找出和修正错误更为容易。我甚至无法确定这些工具能否让我找出所有错误。

编写形式化的规约说明还能训练您把非形式化的规约说明写得更好，因为它帮您考虑得更为周全。能够使用工具来找出设计错误往往是促使工程师开始编写形式化的规约说明的原因。只有写了之后，工程师才会意识到，它能帮助他们考虑得更为周全，进而让他们得到更好的设计。

在开始编码前，我们需要从更高的层级理解我们的编程任务。

描述程序时，我刻画两种东西：它们做什么以及怎么做。编写一段代码时，困难的部分往往是想出代码应该做什么。一旦我们理解了这一点，编码会相当简单。有时候，拟执行的任务需要复杂的算法。我们应该在编码前设计算法，确保算法正确。算法的规约说明描述了代码该如何工作。

但是，有些程序不值得我们详细描述。有些程序写出来是为了学习——或许为了弄清某个规约说明不够详尽的接口——弄清后就扔掉。只有当我们在意程序是否运行正确时，我们才应该详细描述程序。

与思考一样，写作不容易；写规约说明也不例外。规约说明是一种抽象。它应该描述重要的方面，省略不重要的方面。抽象是一种只有通过实践才能学会的艺术。即使我拥有多年的经验，在我理解工程师的问题前，我也无法帮助该工程师编写规约说明。我了解的唯一通用法则是，代码片段用途的规约说明需要描述使用代码时需要知道的所有方面。永远都不应该通过读代码来找出代码用途。

对于哪一个“代码片段”需要规约说明，则没有什么通用法则。就我的编程经历而言，它可能是 Java 类中的字段和方法集合，或是

事件日程

4 月 13 日 -16 日
15 年 CPS 周（CPS Week ‘15）：2015 年信息物理系统周
协办单位：其他组织
华盛顿州西雅图
联系人：Jie Liu，
联系人邮箱：liuj@microsoft.com

4 月 13 日 –17 日
应用计算学研讨会，西班牙萨拉曼卡
主办单位 :ACM/SIG，
联系人：Roger Wainwright，
联系人邮箱：rogerw@utulsa.edu

4 月 14 日 –16 日
HSCC’15：第 18 届国际混合系统会议：计算与控制（CPS 周的一部分），华盛顿州西雅图
主办单位 :ACM/SIG，
联系人：Sriram Sankaranarayanan，
联系人邮箱：srirams@gmail.com

4 月 14 日 –16 日
ICCPS ‘15：ACM/IEEE 第 6 届国际信息物理系统会议（与 2015 CPS 周同时举办），
华盛顿州西雅图
协办单位：其他组织
联系人：Ian Mitchell，
联系人邮箱：mitchell@cs.ubc.ca

4 月 14 日 –16 日
IPSN ‘15：第 14 届国际传感器网络信息处理会议（与 2015 CPS 周位于同一会址）
华盛顿州西雅图
协办单位：其他组织
联系人：Bhaskar Krishnamachari，
联系人邮箱：bkrishna@usc.edu

4 月 18 日 –23 日
CHI ‘15：计算系统中人类因素的 CHI 会议，韩国首尔
主办单位 :ACM/SIG，
联系人：Jinwoo Kim，
联系人邮箱：
create2gether@gmail.com

4 月 21 日 –24 日
EuroSys ‘15：2015 年第 10 届 EuroSys 会议，
法国波尔多
主办单位 :ACM/SIG，
联系人：Laurent Reveillere，
联系人邮箱：reveillere@labri.fr

《ACM 通信》下月预告

机器人动作中运动选择原则的优化

斯诺登事件之后的隐私行为：国家监控曝光的短暂影响

蕴含希望的平行处理

现在没有

解码印度计算机科学中的女性气质

管理您的数字生活

传统编程能否为平行计算应用填补隐身空白?

教授基础性语言原则

以及能耗与准确度之间的取舍，人性化的机器人和数据/科学报道方面的最新新闻。

方法中难以理解的代码片段。对于设计分布式系统的工程师而言，一份规约说明可能会描述一个协议，这个协议由在多台不同的计算机上运行的多个程序中的代码实现。

学校应该教授规约说明。一些大学开设了规约说明的课程，但是我相信，其中的大多数都着重于形式化的规约语言。在他们教授的内容中，与编写真正的规约说明这种艺术有关的任何东西都是意外得来的副产品。规约说明的老师应该为自己的代码编写规约说明，正如教授编程的老师应该自己编程一样。

计算机科学家相信语言中魔幻性质，但对规约说明的讨论会很快转向与规约说明语言有关的话题。有一种用于精确地描述事物的标准语言，它的发展历程延续了几千年：数学。编写非形式化的规约说明的最佳语言是普通的数学语言，它由精确的散文组成，其中融合了数学标记。（有时候，在规定程序如何工作时，编程语言中的其他标记可能也有用。）大多数规约说明需要的数学极为简单：谓词逻辑和初等集合论。对于程序员而言，应用这种数学知识应该像会计师应用数字那般自然。不幸的是，美国的教育体系成功地让大多数程序员害怕哪怕是这么简单的数学知识。

数学的发展并不以被工具检查为目的，而且大多数数学家几乎不知道如何用形式化的方法描述事物。规约语言的设计者们通常转而从编程语言中获取灵感。但是建筑师不会用砖和木板绘制蓝图，规约说明也不应该用代码编写。在我们学到的编程语言知识中，有很多知识不适用于编写规约说明。例如，信息隐藏在编程语言中相当重要。但是，规约说明不应该包含本应被隐藏的，更低层级的详细信息；如果它描述了这些信息，那就是用于编写说明的语言存在问题。我认为，规约说明使用的语言越接近通常的数学，就越能够帮助我们思考。一种语言可能不得不舍弃数学中的某些优雅和能力，以提供有效地工具检查规约说明，但是我们不应该抱有幻想，觉得它正在改进普通的数学。

倡导先编写测试再编写代码的程序员往往认为，这些测试能够作为规约说明。编写测试迫使我们去思考，而任何在编码前让我们思考的东西都有利于我们。然而，编写测试代码并不能让我们跳出代码层级思考。我们可以把规约说明写成该程序应该通过的，高层级的测试描述列表——基本上是程序应该满足的属性列表。但是，这通常不是编写规约说明的好方法，因为很难从中推断出程序在每个场景下应该做或不应该做的事情。

测试程序是一种发现代码错误的有效方式。但它却不是在设计中或在程序实现的算法中找错的好方法。发现这种错误的最好方法是通过站在更高层级的抽象层面思考。通过测试来发现它们就是碰运气。测试不大可能发现只会偶尔出现的错误——但并发系统的设计错误通常具有这一特点。这种错误只能通过证明发现，这往往太难，或者通过穷举测试发现。穷举测试——例如，通过模型检查——往往只能适于某个系统的少量抽象规约实例。不过，它发现错误的效果非常惊人——哪怕是少量的模型，效果也很好。

蓝图的比喻可能会让我们误入歧途。蓝图是图画，但是这并不意味着我们应该用图画描述规约。任何帮助我们思考的东西都有用，图片也有助于我们思考。不过，绘图可能会隐藏马虎的思考。（例如，所有三角形都是等腰三角形的经典平面几何“证明”。）图片往往会

如果我们不从规约说明开始，那么我们编写的每一行代码都是一个补丁。

掩盖复杂性，而不是通过抽象来处理复杂性。它们可能适于简单的规约说明，但是它们却不适于处理复杂性。这是为什么几十年前人们大都放弃用流程图来描述程序的原因。

蓝图和规约说明的另一个区别是蓝图会让人迷失。确保蓝图与建筑物存放在一起一点都不容易，但是规约说明能够且应该作为注释放在其说明的代码的内部。如果工具要求把形式化的规约说明放在单独的文件内，则文件的副本应该作为注释放在代码内。

在真实的场景中，往往不得不在已经制定了程序的规约说明后又修改程序——或是添加新的特性，或是编码时发现了问题。但是，极少会从头开始重写规约说明；与此相反，人们会更新规约说明，修补代码。人们往往会据此争辩，说这让规约说明一无是处。由于下列两个原因，这一论点存在缺陷。首先，修改没有文档的代码无异于一场噩梦。我编写的规约说明提供了非常宝贵的文档，帮助我修改我编写的代码。其次，每个补丁让程序及其规约说明变得更为复杂一点，因此更难理解和维护。最后，除了从头重写程序之外，可能没有别的选择。如果我们不从规约说明开始，那么我们编写的每一行代码都是一个补丁。这样一来，我们在一开始就把不需要的复杂性注入了程序中。正如德怀特 • 大卫 • 艾森豪威尔观察的那样：“没有哪场战役能照搬计划获胜；也没有哪场战役不要计划也能获胜。”

反对规约说明的另一个论点是，程序的需求可能太模糊或定义不清晰，无法精确地用规约说明描述。定义不清晰的需求并不意味着我们不需要思考，而是意味着我们甚至不得不在程序应该做什么方面思考得更多。而且，思考意味着确定规约。当编写 TLA+ 的代码美化程序（pretty-printer）时，我决定不再不加思索地设定公式的格式，而应该用用户想要的方式对齐公式（见附图）。

对齐的样例。

User Input	Naive Formatting	Desired Formatting
`a = b` `/\ ccc >= d`	$a = b$ $\wedge\ ccc \geq d$	$a = b$ $\wedge\ ccc \geq d$

但我无法精确地描述用户想要的结果。我的规约说明由六条对齐规则组成。其中的一条规则是：

如果记号 *t* 是左 - 注释的记号，则它的左注释与它的上覆记号（covering token）对齐。

其中像“*上覆记号（covering token）*”这样的术语有精确但非形式化的定义。正如我看到的那样，通常情况下，这不是编写规约的好方法，因为很难理解一组规则的结果。所以，虽然实现规则容易，但是调试规则却不容易。但是，与理解和调试 850 行代码相比，理解和调试六条规则容易得多。（我在代码中加入了调试语句，报告正在应用哪些规则）由此得出的程序有时候会出错；当程序是否正确依赖于主观判断时，没有程序能不出错。然而，与不编写规约说明相比，编写规约说明后，程序工作得更好，写程序的时间也更短。最近，我增强了程序的功能，使之能够处理某种特殊的注释。有了规约说明后，这个任务变的相当简单。如果没有规约说明，我可能要不得不从头开始重写代码。无论问题的定义可能有多么得不清晰，解决问题的程序都必须做事。我们应该通过考虑问题及其解决方案来找出更好的解决方案，而不是仅仅考虑代码。

反对规约说明的另一个相关论点是，客户往往不知道他们想要什么，所以我们最好尽我们最快的速度编码，这样客户就能告诉我们结果里有哪些错了。蓝图的比喻轻松地驳倒了上述论点。

程序员的主要目标似乎是用更快的速度生产软件，所以我应该得出结论，说编写规约说明节省时间。但是我得不出这样结论。做任何任务时，做得更差或许可以节省时间和精力。当程序员确信规约说明在浪费时间时，强迫程序员编写规约说明可能无济于事——就如同我碰到的很多文档一样。（此处描述了 *TextEditor* 类的 *resetHighlightRange* 方法：“重置该文本编辑器的高亮区域。”）

为了编写有用的规约说明，您必须有编写好代码的意愿——容易理解、运行良好且基本无误的代码。您必须有足够的动力，愿意在编码前花时间思考和制定规约说明。如果您做出了努力，那么规约说明能够节省时间，因为它能在设计错误进入代码前，容易纠正的时候发现设计错误。形式化的规约说明还能让您做出在其他时候不敢尝试的性能优化，因为检查您规约说明的工具能够给您信心，让您相信它们的正确性。

规约说明并没有什么魔力性质。它不能消灭所有的错误。它不能找出编码错误；您还是必须进行测试和调试，以发现这些错误。（语言设计和调试工具在发现编码错误方面进展巨大，但它们却不擅长发现设计错误）。如果需求不正确，那么即使我们已经证明形式化的规约说明满足了所需属性，那份形式化规约说明也可能是错的。思考并不能保证您不犯错误。但是不思考却保证您会犯错误。 C

Leslie Lamport（lamport@microsoft.com）是微软研究院的首席研究员和 2013 年 ACM 图灵奖的获奖者。

译文责任编辑：谢涛

DOI:10.1145/2717517

文章编写主导者 acmqueue
queue.acm.org

最终发现，动态系统的安全性仅仅是低了一点。

PAUL VIXIE

静态与失败之间的抉择

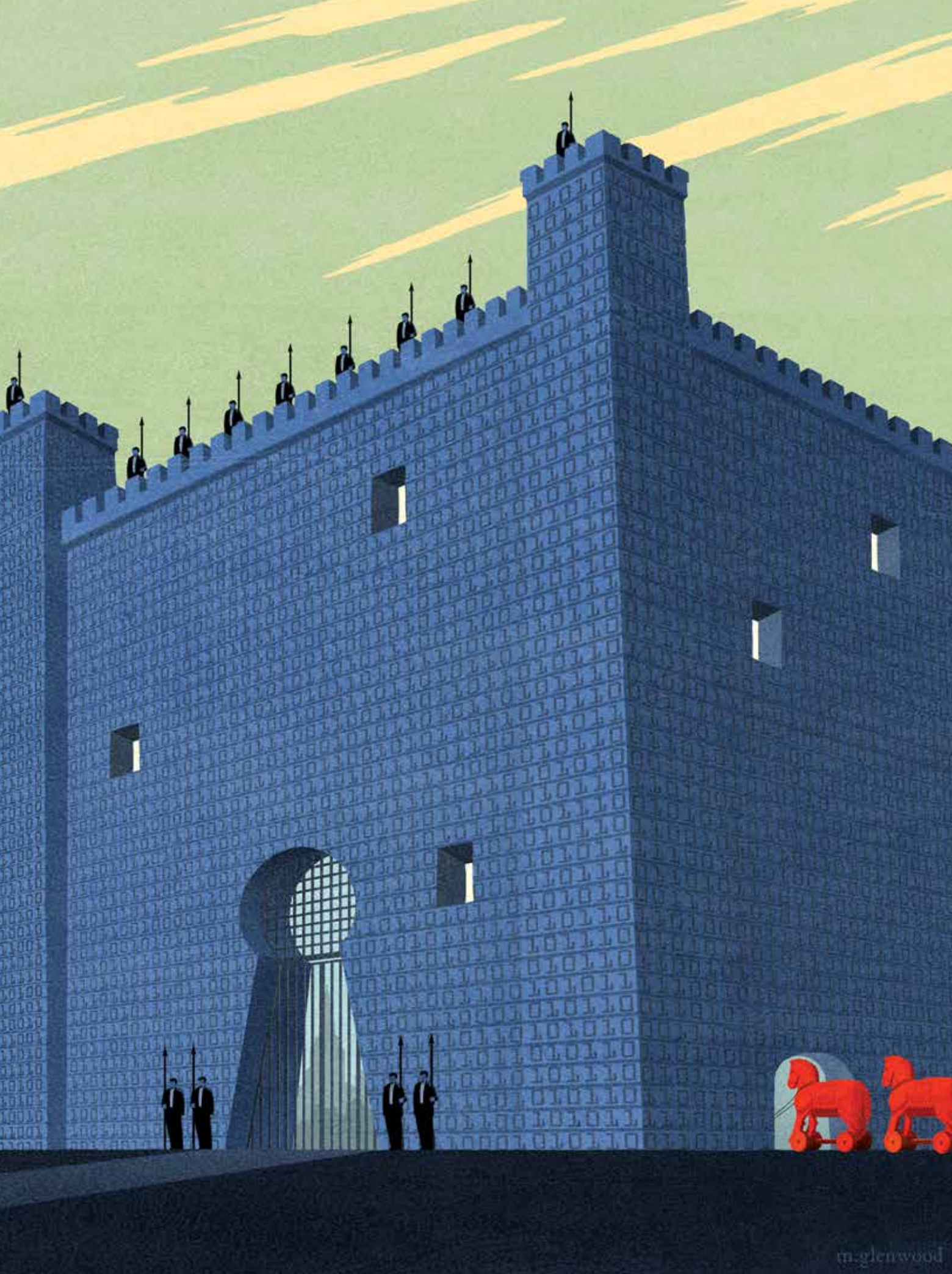
现在和历史上存在的大多数计算机和网络安全问题归根结底均源于一种情况：*让他人控制我们的设备对我们是不利的*。下一次我会解释“他人”和“不利”的意思。在本文中，我会专注于解释我理解的*控制*的含义。使我们失去设备控制权的一种方式是外部的分布式拒绝服务（DDoS）攻击，它让网络中充斥着不想要的流量，从而无法传递真正（“想要”）的流量。其他形式的 DDoS 与此相似——比如，利用低轨道等离子大炮（LOIC）发起的攻击可能不会完全占满网络，但它会让 Web 服务器忙于响应无用的攻击请求，从而无法响应任何有用的客户请求。无论哪种方式，DDoS 都意味着外人正在控制我们的设备，这对我们是不利的。

监控、向外渗漏和其他形式的隐私泄露往往采用了恶意软件或硬件的形式（所谓的“恶意程序”），它们以某种方式进入您的设备，增加一些功能，如读取您的地址簿，或监控您的按键情况，然后把该信息报告给他人。恶意程序提供商往往比作为用户（或制造商）的我们更了解我们的设备，特别是如果他们已经侵入了我们的供应链。这意味着，有的时候我们自认为不能对我们所使用的设备编程，但是实际上了解设备漏洞或秘密握手的他人可以对其进行编程。监控和向外渗漏仅仅是设备在所有者不知情、不愿意和不能控制的情况下偷偷行事的例子。

因为互联网是分布式系统，它需要在设备（如计算机和智能机）之间传递消息，其中每个设备都包含一些硬件和一些软件。到目前为止，最常见的方法是通过发送以某种方式刻意伪装的消息来利用接收设备硬件或软件中的缺陷或漏洞，把恶意程序注入该设备。此时，我们认为是数据的某些东西变成了代码。在可以从其他设备接收消息的设备中，大多数的防御机制均能阻止那些预计包含文本，或图片，或可能是电子表格的数据变身为*代码*。所谓代码是指告诉设备如何运行（或确定其功能）的指令。杀毒软件未能兑现的承诺是，可以使用模式识别检测出恶意程序。如今我们使用杀毒工具来清理被感染的系统，但是我们知道，我们无法通过及时检测恶意程序来预防感染。

所以，我们加固我们的设备，想方设法让外部的数据不变成内部的代码。我们关闭了不必要的服务，我们给操作系统打上补丁，更新操作系统，我们使用防火墙控制谁可以访问我们没有关闭的服务，我们用密码学的方法签署和验证我们的

插图由 MICHAEL GLENWOOD 提供


m.glenwood

代码，而且我们让程序内存中的对象位置随机化，这样一来，如果外部数据以某种方式变成了设备内部的代码，该代码将猜错侵入的位置，所以无法伤害我们。我们把设备内存的一些部分设成仅供数据使用，其他部分仅供代码使用，所以成功的攻击也只能把他人的数据放入非代码区，占据一部分不允许代码执行的设备内存。

我们记录系统的访问日志，防火墙的命中情况以及网络中的流量，想方设法校准正常的情况，以便突显不正常的情况。我们订购网络信誉系统的服务，让已知感染了恶意程序的其他设备无法访问我们的服务。我们在客户注册系统中添加验证码，让僵尸网络无法创建假账户，进而无法依赖假账户从系统边界内部攻击我们。我们把每一个面向互联网的服务放入它自己的虚拟机里面，这样一来，成功的攻击只能接触企业的很小一部分子集。

把特洛伊木马放进来

然后，在耗费心机构建所有复杂的防御设施后，某些人会继续安装动态内容管理系统（DCMS）作为面向公众的 Web 服务器。这种方法就像先建造一座城墙高耸的城市，然后把特洛伊木马放进来，又像创造了一个除了脚跟之外都不会受伤的阿基里斯。在成千数百种 DCMS 中，WordPress 和 Drupal 是其中的范例。DCMS 是用于网站管理的必要的和很好的工具，但是它的位置并不处于把我们的代码暴露给外部数据的前线。

DCMS 的优势在于非技术的编辑能够改变网站或添加网站内容，而那些变化几乎可以立刻让公众或客户见到。在万维网发展的早期，网站是人们通过使用 UNIX 服务器上的文本编辑器，用原始的 HTML 编写的。这也就是说，在网络发展的早期，所有的发布工作都需要有能力利用 UNIX 文本编辑器处理原始 HTML 的技术用户。虽然我个人认为那些是“ the good old days”，但我也承认，如果网络完全由技术用户控制，那么与现在相比，网络会变得更没趣味，成效更低。DCMS 这种系统实现了让网络成为印刷机的前景——它让每个人都成为潜在的出版人。当向公众演讲的能力仅限于有钱人，有权人或技术高手时，人类社会便无法繁荣。

对于使用动态内容管理系统的操作员而言，该系统极其危险。

然而，对于使用 DCMS 的操作员而言，DCMS 极其危险。原因在于，用于编写 DCMS 的计算机语言具有难以置信的能力和灵活性，而且 DCMS 本身也具有强大的能力和灵活性。虽然我们往往失败，但 DCMS 给了我们重新挑起外部数据和内部代码之间战争的机会。大多数用于编写类似 DCMS 网络应用的计算机语言都包含一个名为 `eval` 的功能，利用这个指令人们可以特意把数据中的编程指令提升至运行时的代码。我觉得这听起来很疯狂，而且确实有点疯狂，不过 `eval` 仅仅是体现*所有强大的工具可以杀*人的另一则例子。在技术娴熟的用户手中，`eval` 是创造成功的工具，但如果不熟练或恶意的用户使用它，那么 `eval` 就会变成制造灾难的秘方。如果您想知道攻击者通过 `eval` 它们的数据，进而找到一种侵入您的代码的新方式时有多兴奋和愉悦，您可以在网络上搜索“小波比表（Little Bobby Tables）”。

但是，即便用于编写 DCMS 的底层计算机语言，或是用于管理程序长期数据（比如学生记录）的数据库中没有 `eval`，大多数 DCMS 内部也是由数据驱动的，也就是说 DCMS 软件通常会被构建成类似机器人的东西，它把网站的内容当成必须遵守的指令集合。为了让 DCMS 把数据提升至代码，进而攻击 DCMS，有时候只需要添加一篇格式特殊的博文，甚至也可以通过对现有博文进行评论实现。为了防御 DCMS，抵抗这种类型的攻击，我们需要审计用于编写 DCMS 的每一种软件，包括计算机语言的解释器；所有的代码库，尤其是 Open SSL；操作系统（包括其内核、实

用工具和编译器）；Web 服务器软件；以及任何与 DCMS 一起安装的第三方应用。（提示：这相当可笑。）

分布式拒绝服务

让我们从远程代码执行的漏洞（把外部数据提升至可执行代码）暂时回到 DDoS 上来。即便您的 DCMS 完全没有互动，也就是说它从未给用户提供机会进行任何输入，输入数据的路径使用的 URL 和请求的环境变量也经过仔细审计，而且在运行 Web 服务器的计算机上也没有安装类似 Bash 的环境，DCMS 仍然会是触发 DDoS 攻击的“kick me”标记。这是因为每次 DCMS 的页面浏览均需要在您的 Web 服务器上运行少量的软件代码，而不是仅仅返回之前生成的某些文件的内容。在现代的计算机上执行代码相当快，但与返回预先生成的文件内容相比，它仍然慢了很多。如果某个人用 LOIC 或任何类似的工具攻击网站服务，耗尽 DCMS 资源所需的攻击者数量要比耗尽静态或基于文件的服务所需的攻击者数量少 1,000 倍。

敏锐的读者会注意到，我的个人站点是 DCMS。我不会用类似”鞋匠的孩子不穿鞋“的某些蹩脚借口辩护，而是会指出 DCMS 的优势太明显，即使我也能看到——在不必要的时候，我也不喜欢用 UNIX 文本编辑器编辑原始的 HTML，而且我的个人 Web 服务器不是收入来源，也没有敏感数据。有时我会受到 DDoS 的攻击，而且我也必须定期介入，删除大量垃圾评论。不过，总体拥有成本相当低。如果您的企业网站像我的个人网站那样无足轻重，那么您尽管放心地像我这样运营 DCMS。（提示：在您的企业网站上贴上“踢我”的标识可能会损害您的业务。）

在工作中，我们面向大众的网站是完全静态的。现在有一种内容管理系统（CMS），但是它极为技术化——它需要使用 UNIX 文本编辑器，使用名为 GIT 的版本控制工具，并了解名为 Markdown 的语言的知识。它让我们的非技术雇员（包括业务团队的某些成员）感到失望，但是它意味着我们的 Web 服务器在渲染 Web 对象时不需要运行代码——它只是返回用 Ikiwiki CMS 预先生成的文件。Bricolage 是非动态 CMS 的另一个例子。与类似 Ikiwiki 的系统相比，它对非技术的 WYSIWY 用户要友好一点。请注意，没有系统能够防止 DDoS，虽然它们的市场材料或年报可能说得很好。我们都生活在缺乏任何访问控制的互联网中，所以大多数低投入攻击者可以轻而易举地击倒大多数高投入的防御者。不过，我们确实可以选择我们的网站是否贴有“踢我”的标签。

还有一种混合模型，我把它称为 *大致静态*。在这种模型中，在各视图中均不变化，且能被多个浏览者共享的所有样式表、图片、菜单和其他对象是预先生成的，作为文件提供。在浏览者登录之前，甚至是登录之后，Web 服务器都不会以浏览者的名义执行任何代码，每次页面浏览返回的大多数对象是静态的（从文件中获取）。与*完全静态*的网站相比，它的安全性稍低一点，但对于很多 Web 服务的运营者来说，它是切实可行的折衷办法。我说“安全性稍低”的理由是，攻击者可以在服务内注册一些账号，以便提高他们随后的攻击效果。对于僵尸网络而言，大规模创建账号是一项常见的任务，所以大多数允许在线注册的 Web 服务运营方试图使用验证码（CAPTCHA）来保护他们的服务。

大致静态的模型也能与内容分发网络（CDN）一同使用，其中浏览者的网络浏览器实际连接的前端服务器位于云中，该服务器由专家操作，配置超高，可处理除最高等级的 DDoS 之外的所有 DDoS 攻击。为了实现这一功能，网站必须标出静态对象（如图片、样式表和 JavaScript 文件）可以缓存。这告诉 CDN 的提供商，它可以把这些文件分布在它的网络中，向很多浏览者返回很多次文件——在面临 DDoS 时，向很多攻击者返回很多次文件。当然，一旦用户登录进入了网站，会有一些动态的内容，这时 CDN 会把请求传给真正的 Web 服务器，而 DCMS 会再次被暴露给外部数据。这永远都是制定关注、警惕、警告和应急计划的原因。

对于混合式的，大部分基于 CDN 的模型，可以把大致静态的 DCMS 放在前端 Web 代理（如 Squid 或 Apache 的 *mod_proxy* 特性）的背后。这种方法不能像外包给 CDN 一样很好的保护您的网络不受 DDoS 攻击，但是它能够保护您的 DCMS 资源不被耗尽。请注意，任何大致静态的模型（有 CDN 或无 CDN）仍然无法保护您的 DCMS 代码，让它不受他人数据的影响。对于在安全行业工作的大多数人来说，这意味着，如果静态的发布模型可以满足 Web 服务的业务目标，那么静态比大致静态要好。

所以，如果您想认真运营基于 Web 的服务，不要把“踢我”的标记贴在上面。静态与失败之间的抉择 C

queue.acm.org 上的相关文章

Finding More Than One Worm in the Apple
Mike Bland
http://queue.acm.org/detail.cfm?id=2620662

Internal Access Controls
Geetanjali Sampemane
http://queue.acm.org/detail.cfm?id=2697395

DNS Complexity
Paul Vixie
http://queue.acm.org/detail.cfm?id=1242499

Paul Vixie 是 Farsight Security 公司的首席执行官（CEO）。此前，他曾担任过 ISC（互联网系统协会）会长、主席和奠基人；MAPS、PAIX 和 MIBH 的主席；以及 Abovenet/MFN 的首席技术官。

译文责任编辑：陈贵海

DOI:10.1145/2735589

使用军事术语后，用户可以创建标签和绘制符号，从而把对象定位在数字化的地图上。

PHILIP R. COHEN，EDWARD C. KAISER，M. CECELIA BUCHANAN，SCOTT LIND，MICHAEL J. CORRIGAN，R. MATTHEWS WESSON

Sketch-Thru-Plan 指挥和控制使用的多通道界面

2000 年，OVIATT 和 Cohen[25] 预测，多通道用户界面将“增强并最终取代目前在许多计算机应用中使用的标准化的图形用户界面”。多通道交互往往聚焦于采用了多种可选择输入通道的移动界面，包括语音、触摸和手写以及基于地图的界面，设计用于处理和融合多种同步的通道。在过去的几年中，采用多种可选择输入通道的基本多通道界面确实成了移动设备的主流界面。在本文中，我们描述了一个基于融合方式的高级多通道地图系统。该系统名为 Sketch-Thru-Plan（STP）。2009 至 2011 年在美国国防高级研究计划局深绿（DARPA Deep Green）计划下支持下研发，支持在指挥和控制（C2）时快速创建地面军事行动的作战方案。为了更好地了解背景知识，我们探讨了这种挑战来自于 C2 的地面行动系统和用户界面。我们讨论了 C2 的图形用户界面如何造成军队行动效率低下，培训成本高昂。为了处理这些问题，我们接着阐述了 STP 的多通道界面以及评估。最后，我们讨论了本系统在美国陆军和美国海军陆战队中的部署情况。本次的案例研究涉及以用户为中心的设计和开发过程。有前景的基础研究需要这种过程以实现可靠伸缩，进而融入大型组织中的任务关键产品。

指挥和控制

指挥和控制软件必须满足指挥官和各类参谋的需要，这些军官的范围很广，涵盖从高级指挥官（比如陆军师级或旅级指挥官）、他们自己的专业参谋到经验相对欠缺的，较小单位的指挥官等多个层级。[1] 在整个范围内，士兵迫切地需要那种能在实际和模拟行动使用，同时也能在数字基础设施和计算设备各异的野外和移动场景下正常工作的规划工具。现在，尚无 C2 系统满足了所有这些要求，部分原因是由于 GUI 的限制。

在引入数字系统之前，C2 的功能通过纸质地图实现，当时使用了透明的塑料覆盖物和油彩笔。用户通过与他人交谈进行协作，一边在地图的覆盖物上挥笔，一边制定计划。这种界面的优点是，它不需要举办任何的界面培训，而且操作完全避免了故障。然而，它也存在

» 重要见解

- **多通道界面支持用户把注意力集中在手上的任务上，而不是工具上。**
- **创建和定位符号时，使用了标准化的符号名称和形状的多通道语音和草图界面（speech-and-sketch interface）是一种效率高得多的手段。**
- **较指挥和控制以及规划常用的图形用户界面相比，几乎所有参与测试的人员都更喜欢多通道界面。**

插图由 JUSTIN METZ 提供


STP
9:33
•579,1
550

明显的不足，比如需要把数据复制到数字系统中，缺少远程协作。为了处理这些问题，现在的 C2 系统基于 GUI 技术。使用最为广泛的陆军 C2 系统名为未来指挥所【Command Post of the Future（CPOF）】[11]，它是一种拥有三块屏幕的系统，依赖拖放方法来操纵信息。它支持通过人 - 人对话和协作绘图进行同地协作和远程协作。与之前的 C2 系统相比，CPOF 是一大进步[a]，而且它也成了在 2003 年发起的伊拉克自由行动（Operation Iraqi Freedom）中主要使用的陆军 C2 系统。

表 1 勾勒了 CPOF 用户如何派遣单位在 0800 至 1600 的时间段内沿指定的路线巡逻。熟练的用户可能需要花一分钟时间执行这 11 个高层面的步骤，妥善制定详细计划时，还需要使用很多其他的功能。相比之下，在 2006 年，使用与 CPOF 紧密集成的 Quickset 多通道界面后，用户可以说“Charlie company, patrol this route <draw route> from oh eight hundred to sixteen hundred（Charlie 连 队从 0800 到 1600 巡逻该路线 < 画线 >）”。只用简单地说句话，所有的属性值就会自动填充，在平板 PC 计算机上六秒便可处理完成。

士兵必须掌握很多功能在菜单系统中的位置，如何通过“ctrl- 拖动”[b] 功能的渲染图来连接信息，以及如何在各种屏幕和窗口之间切换。由于使用 CPOF 完成一个标准功能时需要太多的原子 GUI 步骤，SRI 国际（SRI International）和通用动力公司（General Dynamics Corp.）构建了一套学习示范系统，有经验的用户可以使用这套系统描述较高层级的步骤。[21] 在陆军单位内部，专家用户会得到培训，以创建这种步骤，存在的步骤也会作为运营系统的“知识”的一部分传递给单位的其他人员。然而，如果界面支持用更简单的方式表达用户的意图，那就能够减少让系统去学习较高层级步骤的需求。在陆军的校舍和系统部署的场所中，每年都有成千上万的士兵接受培训，学习如何操作这个复杂的系统，开支巨大。

定位资源是一个关键的 C2 计划任务，这个任务通过在地形图上用多个符号来实现。这些符号用于表示作战单元，每一台装备，路线，战术的界限，事件以及任务。标记的名称和形状是军事“条令”或标准化程序、符号和语言的一部分。这样一来，人们能用相对清晰地方式传递意图。士兵花了大量的时间来学习条令，而且任何加强条令的事物都被军队视为极为有利。

每个单位的符号都有一个框架和颜色（说明友好，敌对，中立和联盟），顶部有一个梯队标记（比如排），侧面（两侧）有一个标签或“指示符”，中间标有“角色”（比如装甲，医疗，工程和固定翼航空器），还有数不清的其他标记（见图 1）。这是一种组合语言，通过它人们可以生成几千上万种符号配置。为了使用 GUI 技术来处理数量庞大的词汇，C2 系统往往使用很长的下拉菜单列出可以在地图上放置的各类实体。常见的符号可以排列在控制板上，供用户选取。然而，

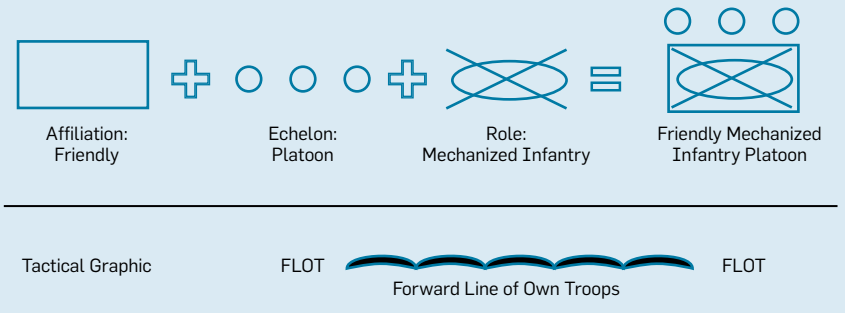
a http://www.army.mil/article/16774/Command_Post_of_the_Future_Wins_Outstanding_US_Government_Program_Award/

b “Ctrl-dragging（Ctrl 拖拽）”指在按下 CTRL 键的同时，在地图符号上按下鼠标左键，然后拖动符号至用户界面上的其他位置；符号的“clone（克隆）”出现在终点位置上，如果原始的符号出现了变化，该克隆也会发生变化。

表 1. 派遣单位通过未来指挥所巡逻的步骤

步骤	CPOF
1	右键单击显示的背景（不是地图），系统弹出可被创建的 CPOF 对象菜单。
2.	选择“任务”，打开一个小窗口；
3.	输入任务标签“巡逻（Patrol）”；
4.	在“执行者（Performer）”槽中输入单位的名称（比如 C/1-62/3-BCT）；
5.	使用鼠标用数字墨迹在地图上描绘路线；
6.	按下 control 键，选择刚刚绘制的数字墨迹（使用鼠标左键），然后把它拖动到任务窗口中的容器盒内。
7.	Ctrl—拖动任务窗口到该链接附近的地图上，把任务的符号放在地图上；
8.	定位和显示“Schedule（调度）”窗口；
9.	Cntrl- 拖动任务至 Schedule（调度）窗口，把 Task（任务）放到与 Performer（执行者）槽中的单位名称相关联的线上。
10.	移动该任务图形到考虑中的日期上；以及
11.	拖动左侧和右侧的时间间隔边界，让它们分别与 0800 和 1600 对齐。

图 1. 组合性质的军事单位符号和战术图形示例



这些控制板可能会变得相当大，从而占据了珍贵的屏幕空间。那些空间用来展示地图、计划和日程安排会更好。

GUI 中使用的另一种识别军事单位方法需要用单位名称、角色、梯队和兵力等属性和属性值来确定它的组成部分。展示时，每一个单元都配有多个更小的菜单，用户可以从中选取值。用户可以在搜索框中输入数据，通过字符串匹配找出可能的单位。但用户仍然需要选择期望的实体，并通过菜单设置属性值。创建或找到符号后，可以通过拖放操作把它定位到地图上。由于系统设计上的这些限制（以及很多其他的限制），用户告诉 STP 的开发人员，基于这种经典 GUI 技术的 C2 系统界面很难学习和使用。我们已经发现，创建和定位符号时，加入了条令语言或者标准化的符号名称和形状的语音和草图界面（speech-and-sketch interface）是一种效率高得多的方式。

基于地图的多通道系统

很多项目都调查过基于地图的多通道交互方式，有的使用了笔和声音的，[3,5,6,14,20,23]，有的使用了手势和声音。[2,7,16,19] 其中的有些系统成了当前成果的研究基础，不过据我们所知，没有任何系统在 C2 中部署。除了智能机之外，最广泛部署的多通道系统是微软的 Kinect，它追踪用户的身体运动并支持语音控制，主要用于游戏和娱乐应用；企业和健康应用也正开始出现；[22] 同时，人们开发了其他的商用多通道系统用于满足仓储需要，并开始在汽车中使用。[26]Adapx 的 STP 系统与二十世纪九十年代后期俄勒冈研究院开发的 QuickSet 系统关系最为密切。[c] Quickset[5,6,14,23–25] 是一个语音 / 草图 / 手写多通道界面的原型，用于基于地图的互动。因为语音处理不需要屏幕空间，所以它的多通道界面可以轻易地部署在平板电脑、PDA 和可穿戴设备以及墙壁大小的显示屏上。它提供了分布式的多人协同操作，可通过语音和 / 或绘制把实体定位在地图上，并可为各实体创建任务，这些任务可以通过模块化的半自动兵力系统（ModSAF）[4,8] 模拟器模拟。QuickSet 也被用于手势控制的三维可视化 [7] 以及大量的联网设备（比如 TV 监视器和增强现实系统 [16]），人们通过声学、电磁和视觉的方法跟踪这些手势。

在对以用户为中心的设计进行广泛的研究之后，俄勒冈研究院的团队证明，在操纵地图时，用户更喜欢多通道的交互方式。他们还能够选择符合其场景和任务的最佳通道或通道组合。[23,25] 用户的草图一般会提供空间信息（比如形状和位置），语音则提供了标识和其他属性的信息。该用户界面模仿了在进行数字化之前的军队使用的纸质地图 [6]，从而减少了用户的认知负担。[23]

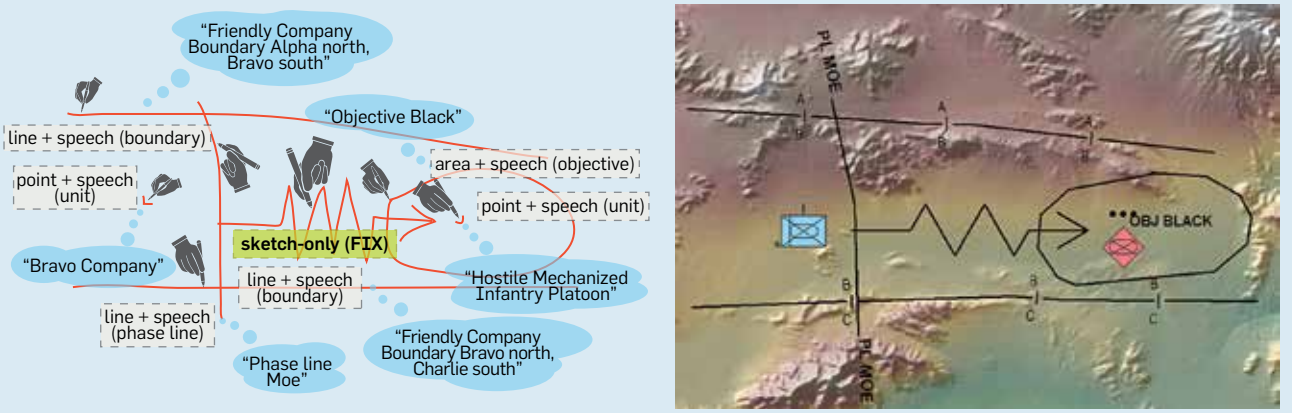
QuickSet 的总词汇量约为 250 个单元符号以及约 180 个“战术图形”，如图 1 所示。语音识别基于早期的 IBM 识别器，草图识别涉及稍加训练的神经网络和隐马尔可夫模型识别器。大部分的研究工作投入到了创造用于多通道融合处理的创新方法上面。[14] 在 QuickSet 中，基于归一的多通道输入融合 [14] 支持各通道的互相消歧（MD），[16,24] 其中系统对某个通道中传递的信息的处理能弥补其他渠道的错误和歧义，进而把相对错误率降低 15%–67%；例如，如果用户说了“boats”而不是“boat”，那么系统便能够消除有三个物体的草图中的歧义。MD 能够提高系统容忍识别错误方面的鲁棒性，这在高噪声环境中相当关键。在高噪声环境中，用户的口音很重，或者草图是在用户移动时创建的，或者创建草图时用户的手臂累酸了。[18]QuickSet 展示了一种在上述现实世界条件下仍能鲁棒工作的多通道界面，这是在野外部署的必要和先决条件。

Sketch-Thru-Plan

2008 年，DARPA 设立了深绿（Deep Green）项目，旨在让规划者在任务规划时使用模拟推演出当敌人

c Adapx 公司是从俄勒冈州波特兰俄勒冈健康和科学大学（Oregon Health and Science University）下属俄勒冈研究院分出的公司。

图 2.STP 把语音和草图（左侧）处理成右侧的数字对象。



采取了最可能和最危险的行动方案（COA）时，我方采取某个行动方案（COA）的结果。COA 是一组实体在某段时间内为完成任务而实施的行动的规格说明。深绿工作的关键部分是开发一种易用的界面，它能支持规划团队迅速创建 COA。DARPA 选择了 Adapx 的多通道技术（源于 QuickSet）以及 SAIC 公司（http://www.saic.com） 和 BAE Systems （http://www.baesystems.com）来完成这个任务。最后，作为总承包人，Adapx 根据主题专家的指导开发了 STP 系统，ROTC（预备役军官训练团）中的学生进行了测试。STP 用户通过语音和绘图编制他们的计划，如果需要，还有可选的 GUI 输入。STP 可以与现有的 C2 系统（尤其是 CPOF 和 LG-RAID 模拟器 [27] 以及基于谷歌地球的可视化模块）实现互操作，把规划中的实体位置和任务传入那些系统。

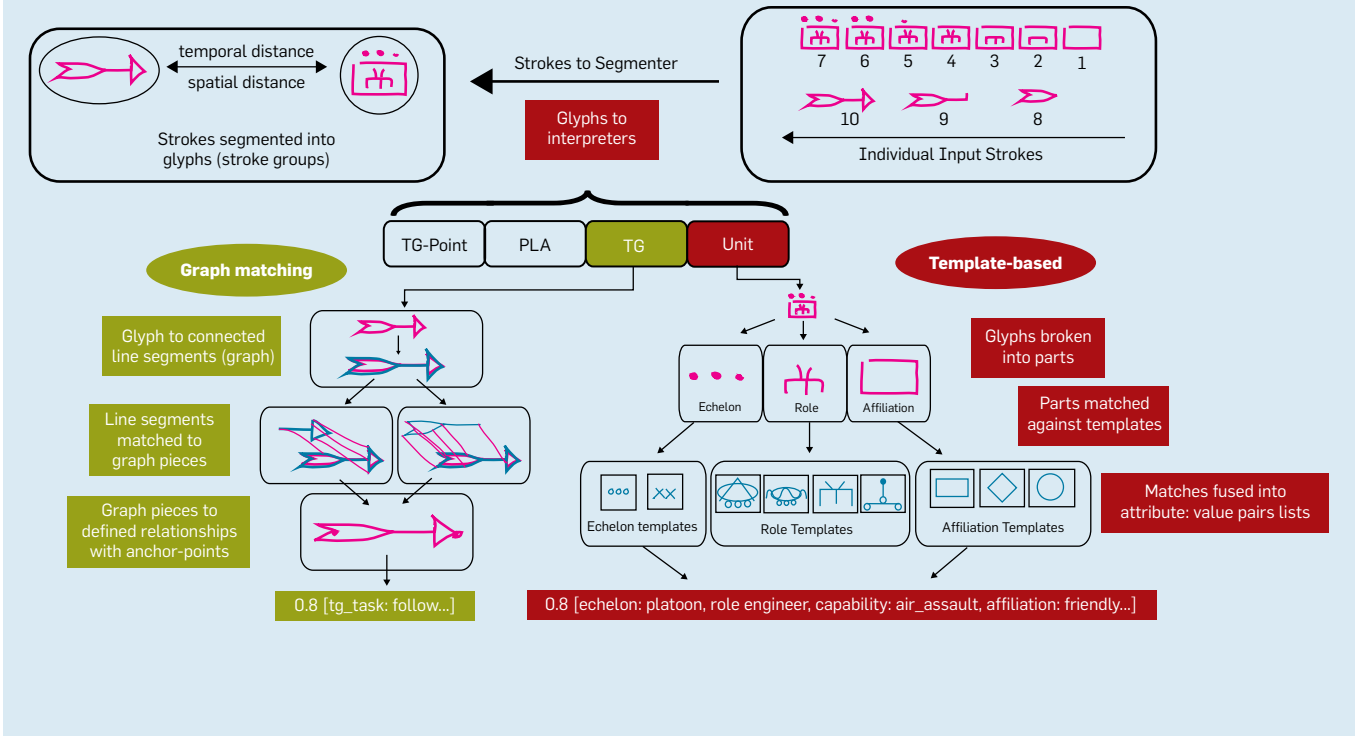
草图的输入通过触控笔、手指、鼠标或数字笔以及纸质输入捕获。大多数人更喜欢同时使用语音和草图，但是单独用草图也非常有用，特别是使用数字纸和笔的时候。最快和最容易的输入方式是绘出简单的点、线或区域作为符号应该放置的位置，同时说出符号的特征（比如状态、联盟、角色、兵力和番号）；例如，在图 2 中（左侧面板），用户可以说出单位的类型或战术图形——“敌对的机械化步兵排”，“目标黑色”，“连队的界限，A 连北面 B 连南面（alpha north bravo south）”——同时提供一个点，一条线或圈出一片区域。用户还能绘制符号，比如“Fix”符号，或图 2 中的折线。系统识别会识别出这种多通道输入，然后融合它们以提供军事符号，如图 2 右侧面板所示。由此得出的符号不仅是位图上的图标，而且还是用数字化的方式记录了地理数据的对象，它们是系统数据库的一部分，可以传入 C2 和仿真系统。

STP 多通道界面可识别 4,000 多个符号和战术图形组态，每一个都有一组属性和值，如图 1 所示。它还可识别 150 多个战术任务（比如沿某条路线巡逻和投递补给），这些任务利用了空间和语义相关的符号组。这些符号通过规定了相关标签、图标和唯一识别符的数据库同时提供给语音和草图识别器。识别器的词汇会自动填充，它支持系统采用其他符号或转向新的领域。

STP 支持一组用户协同编制一个计划，其中不同的用户承担各种功能角色（比如指挥官、后勤人员、情报人员和工程人员），为整体计划做出自己的贡献。系统支持用户创建名义实体（比如默认的步兵连）或定位现有军事组织的实际实体，安排和同步任务，填入工作表数据（这些数据表可用于填充特定角色的系统，比如后勤和工程使用的系统）以及创建所需的文件（比如行动命令）。

与 QuickSet 相比，STP 词汇的范围以及任务的覆盖范围要高一个数量级。它还使用了能力更强的

图 3. 草图识别处理过程概览：TG = 战术图形，PLA = 点 - 线 - 区域。



语音识别，草图识别和匹配子系统。[d] 与 QuickSet 类似，STP 支持跨多个形状因子的同一种多通道界面，包括手持和平板电脑以及数字笔和纸。[6]STP 的设计得到了领域专家的指点，从而得到了功能完整的规划工具。STP 不只是一个研究原型，我们开发出的 STP 技术成熟度较高，[e]，且得到了实际军事作战单位的测试。因此，STP 现在正在被部署到美军内部多个组织中。

STP 组件

下面各节描述了系统的主要功能组件。

语音和自然语言。STP 中口语处理的目的是支持多通道语言，与仅支持语音的构件相比，它更简短，更简单。[25] 它的目的是显著提高用户的绩效，对用户透明，让用户更容易记住，并尽量降低认知的负担和识别错误。STP 的方法如下：用户通常会说出名词短语和画出符号，以创建和定位实体。如之前所述，基本的口语名词词汇及其属性和值定义在数据库里。

当实体的创建涉及多个属性时，我们不鼓励用户用一次长的言语说出所有的信息，因为它会造成复杂的句法，“不流利“和其他的障碍，影响口语互动的健壮性。与此相反，用户可以通过在实体上画一个标记来选择这个实体并随后说出属性 - 值信息 [f]；例如，单位的兵力可以通过标记该单位并说出”减少兵力“予以改变。与此相似，用户可以使用“R O Z <draw area> from oh eight hundred to sixteen hundred（从 0800 到 1600 的限制活动区 < 绘制区域 >）”，”<mark the ROZ>“minimum altitude one thousand meters maximum two thousand meters（< 标记 ROZ> 最低高度 1000 米，最高 2000 米）”来创建特定时间段和高度的限制飞行活动区。对于大多数 C2 系统而言，创建这种三维的战术图片非常耗费时间。

STP 多通道界面可识别 4,000 多个符号和战术图形组态，每一个都有一组属性和值。

与 QuickSet 不同，STP 包括两个自然语言解析器，一个只处理名词短语，它的组成短语可以按任何顺序出现；另一个是 Gemini[9]，它使用了大量的英语语法知识。名词短语解析器在符号创建时使用，用于分析诸如“预计敌人的机械化排兵力有所减少”这样的短语。覆盖范围更宽的 Gemini 解析器在描述实体任务时使用，它通常会涉及动词短语【比如“Resupply along MSR alpha（沿 MSR A 进行再补给）”】. 很多系统中使用了 Gemini，包括 NASA 的系统 [10]，它是设计用于口语系统的，能力最强的英语解析器之一。语法中的动词短语源于一个“任务签名”表，该表规定了每个军事任务必选的和可选的参数类型。因为系统能推断出可能的任务，这尽可能减少了用户说出复杂句的需要。

语音识别使用了 Windows 7/8 中的微软语音引擎，它使用了基于语法的识别。系统的语音识别一直处于激活状态，，但是它必须与触屏配合工作，所以，在不进行草图交互时的人和人之间的谈话不会引起虚假杂乱的识别。STP 会协调多个同时存在的识别器实例，其中每个都有不同的语法，或“上下文”，它们是用户界面状态的函数。上下文知识限制了可能的语音和语言，因此提高了准确度和速度；例如，当人在地图上的某个物体上画一笔时，“属性 - 值”的语法上下文被触发了。因为设置上下文的行为本身可能是模糊不清的，STP 的设计支持比较多个同时存在的识别器的结果，这些识别器体现了不同的限制。

d 与 QuickSet 使用的标有地理信息的位图不同，STP 使用了 ESRI 提供了 ArcGIS 地图系统。

e 有关 NASA 对“技术成熟度等级”的定义，请访问 http://esto.nasa.gov/files/trl_definitions.pdf。QuickSet 通过技术成熟度等级 3 来开发，而 STP 已经通过等级 6 来开发。持续的开发和部署会让它达到等级 9。

f 请注意，虽然通过标记来“选择”并不是一个原子操作，但是它必须被识别和解释，因为笔画可能是模糊不清的。通过标记来选择还避免了一个“模式化”界面，其中选择是一个独立的模式；与此类似，STP 不使用需要硬件协助的选择（比如笔或键盘上的特定按钮）来支持简单的触摸或数字笔和纸。

未来，使用口语听写可能会有所帮助，如 Google Voice（谷歌语音）、Nuance Communications’s Dragon Dictate、Apple’s Siri（苹果的 Siri）以及语音到语音的翻译系统[13]。它们需要开发大规模的统计语言模型。然而，因为构建此类语言模型的军用口语数据可能是机密的，这种创建语言模型的方法可能有问题。因为 STP 可以利用用户的军事术语知识和结构化的规划过程[g]，所以到目前为止，基于语法的语音识别获得了成功。

草图识别。STP 的草图识别器基于计算机视觉的算法，也就是 Hausdorff 匹配。[17] 它使用了一组墨迹解释器来处理草绘的符号和战术图形（见图 3）。对于单位的符号，识别器的算法使用了线段的模板，将草绘的数字墨迹与它们进行比对，然后应用基于笔画距离和笔画角度的，经修正的 Hausdorff 度量来计算相似性。对于战术图形，识别器创造了由符号片段组成的图形，然后把它们与输入进行比对。它们的基础都是空时笔迹分割器。在空间分割方面，如果给定笔画与现有已分割的笔画组（或“字形”）的最短距离低于与现有字形大小成比例的阈值，且它的开始时间距上一笔的结束时间处于用户可设置的阈值内，则将新输入的一笔加入当前的字形。

在解释基于模板的单位图标时，首先定位了附属“框”，然后把字形分割为其组成部分，包括协同单位、角色和梯队，它们具有相对于框的标准位置。虽然各种角色本身都可能具有内在的组织结构，但它们作为一个整体进行匹配。如果诠释图标的语言内容与通常期望的内容一致，则使用微软的识别器处理手写字符。然后，把这些部分与模板图片库进行比较，再综合结果，得出识别的输出。如果没有发现符号的“框”，那么草图识别器会试着使用战术图形解释器。对于战术图形（它的形状可能被拉伸或扭曲），算法使用了图匹配方法，它首先把字形分割成了由线段和节点组成的图。然后，把该图与允许拉升或扭曲的，分段的图模板进行匹配。再根据草图的规则重新组合这些分片。这些规则定义了分片和锚点之间的关系，根据这些关系可以构建一个完整的符号；例如，这种规则把“forward line of own troops（FLOT）（本部队的锋线）”符号（如图 1 所示）定义为半圆的“线阵”（一个“原语”），这个“线阵”带有由两条大致平行的线组成的铁丝网围栏和一个平行的，由多个圆组成的线阵。

该界面利用和增强了士兵已经拥有的技能，因为他们已经接受过标准化语言、符号和军事决策过程方面的培训。

用于单元图标识别的基于模板的方法的优点是，它可以通过向库中添加新模板来轻松扩展；例如，新的单元角色能以可扩展的矢量图形的形式加入，然后组合单元符号识别器会把它定位在附属的边界之内。

明示和暗示的任务创建。除了在地图上创建和定位符号外，用户还能明示地说明任务，或依赖系统来暗示地为使用当前符号的任务集构建递增性解释 （见图 4）。通过把地图上的符号与可能的域任务（比如战斗服务单元执行“补给”和医疗单元执行“撤出伤兵”）的参数类型进行匹配，STP 实现了后一种推断，但其受时空约束的限制，如图 4 所示。STP 向规划者提供了实时可视化，用于展现正在创建的计划中各种任务的匹配情况。规划者可以轻松地检查潜在的任务，根据需要接受或改正任务。在此处，STP 推断出，战斗服务支持（Combat Service Support）单位和主补给线（Main Supply Route）A 被组合成一个沿主补给线 A 的再补给（Resupply）任务。如果这种推断是正确的，规划者可以选择复选框，然后会更新任务矩阵和调度。随着规划者在地图上添加更多的符号，系统对匹配任务的解释也会随着更新。任务的开始和结束时间可

g 军队已经教了士兵如何使用结构化的军事决策过程。[28]

以用语音，或在标准甘特图任务同步矩阵的图形上进行调整。请注意，STP 不会尝试进行自动规划或计划识别，而是在规划过程中提供协助；例如，STP 可以从任务和图形中生成模板式的“作战命令”，这是规划过程所需的输出。从原则上来说，还可以提供更多的规划协助，虽然我们并不清楚规划者更喜欢什么。

因为系统是由数据库驱动的，多通道界面和系统技术拥有很多潜在的商业用途，包括其他类型的运营规划（比如消防员口中的“林野”灭火）以及地理信息管理，计算机辅助设计和施工管理。

评估

美国军队进行了四种类型的 STP 评估：组件评估，用户评判委员会（user juries），对照研究和规划练习方面的测试。

组件评估。2008 年，在深绿计划中，人们通过 DARPA 选择的第三方评估器进行了 172 个符号的识别测试，根据能否识别位于潜在的符号 - 识别假设列表顶部的正确值这一标准，STP 的草图识别算法的准确度为 73%。依据高分假设（top-scoring hypothesis），效果位居第二的是深绿（Deep Green）草图识别器，它针对相同的符号构建，与 STP 使用了相同的数据并在相同的时间进行了测试，识别准确度达到了 57%。[12] 大多数用户不止使用了草图，他们更喜欢多通道交互，在绘制点、线或区域的同时说出标签。2008 年，根据外部承包的评估器的报告，STP 的多通道识别的准确度明显高于 96%。如果 STP 的解释不正确，用户通常能够重新进行多通道输入，在其他的推测符号列表中选择符号，或调用多通道帮助系统。该帮助系统会说明系统的范围，并可用于创建符号。

人们还使用了头戴式消噪麦克风测试了在高噪声美军车辆中 STP 的性能。两名用户——一男一女——每人均在野外乘坐处于运动中两种类型的车辆，其中平均噪声为 76.2dbA，峰值达 93.3dbA，两人共发布了 221 条多通道命令。在实验室中，把录制的车辆噪声放到最大音量（均值为 91.4dbA，峰值为 104.2dbA）时，他们向 STP 发布了相同的 220 条多通道命令，在这些测试中，多通道识别准确度的结果分别为 94.5% 和 93.3%。我们猜测，除了多通道架构外，消噪麦克风可能也抵消了声大但相对恒定的车辆噪声。在后续研究中，STP 团队期望在更大的研究中分离这些因素的贡献。

用户评判委员会（user juries）。美军测试软件的一种方式是，让在海外驻扎后刚刚回国的士兵加入一个“用户评判委员会”，试用潜在的产品，并提供意见说明它能否在他们最近的活动中提供帮助。为了获得士兵对 STP 的反馈，2011 年 -2013 年期间，美国陆军训练与条令司令部（Army’s Train-

图 4. 暗示的任务创建。

该地图描述了战斗 - 服务 - 支撑单位和主要补给线路，医疗单位，以及伤亡收集点；该任务推断过程找到了多个任务（比如再补给），这些任务可能包括主要补给线路 A 沿线的各种实体。

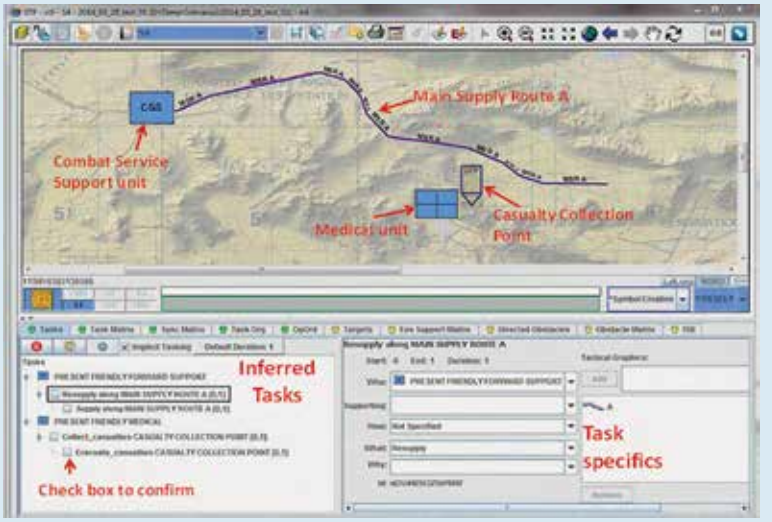
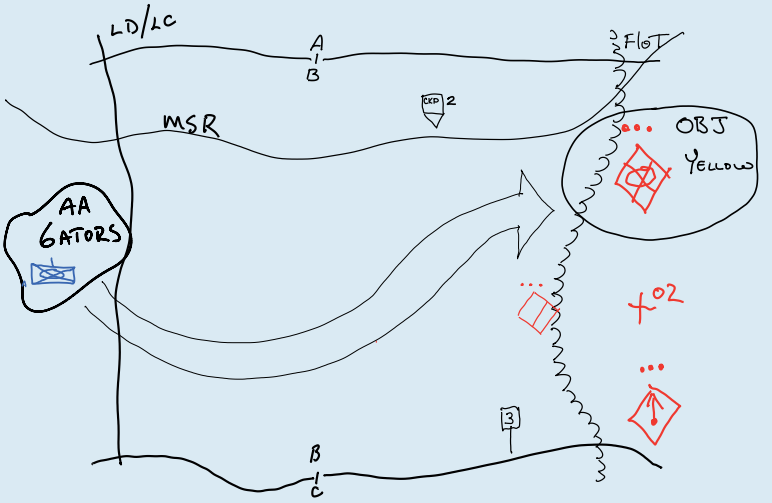


图 5. 对照研究中使用的 COA 草图，2013 年 1 月



ing and Doctrine Command）邀请了来自四个陆军师的 126 名士兵对 C2 系统和 / 或 CPOF 与 STP 进行比较。这些士兵都有车载 C2 系统和 / 或 CPOF 系统的使用经验。为了保密，本文改变了这些师的名称，把它们简单地称为 1 师，2 师，3 师和 4 师。STP 的开发人员培训了士兵 30 分钟有关 STP 的知识，然后给了它们一份 COA 草图，要求他们用 STP 输入该草图。然后，士兵们填写了一份五分制李克特式问卷。与之前使用的 C2 系统相比，在所有的项目中士兵都认为 STP 更好用，也更喜欢 STP；表 2 汇总了士兵对 STP 与之前 C2 系统的对比评分。

对照的用户研究。承包商很难针对现役士兵进行对照研究。然而，在 2013 年 1 月举行的 STP 用户评判委员会中，来自 3 师的 12 名有经验的 CPOF 用户在一项可控试验中对比和评估了 STP 与 CPOF。实验使用了系统使用顺序的影响已被抵消的被试内设计。在实验中，STP 开发团队对有经验的 CPOF 用户，进行了 30 分钟的 STP 培训，然后给了他们图 5 中的 COA 草图，要求他们通过 STP 和 CPOF 两种方法输入草图。结果说明，使用 STP 的多通道界面后，这些经验丰富的 CPOF 用户在地图上创建和定位单位以及战术图形的速度比使用 CPOF 要快 235%[h]；表 2 中列出了这些实验对象的评论。请注意，“放置符号”只是规划过程中的一个步骤，规划过程还包括制定各单位的任务以及创建一个完整的 COA 和一个作战顺序。专家们报告，执行上述其他的规划功能时，STP 节省的时间明显更多。

规划练习方面的测试。最近，一组承担了开发 COA 练习任务的专家规划者使用了 STP 来完成任务。这些 COA 练习最终会出现在 CPOF 上。STP 团队与使用微软 PowerPoint 的专家规划团队（他们较不喜欢 CPOF）在一起工作，开发计划。很多人尝试过用各种规划工具（包括 CPOF 本身）来开发 COA 练习。不过，尽管 PowerPoint 有很多缺陷（比如地理空间的逼真度不高），他们仍在继续使用 PowerPoint，因为大家都了解 PowerPoint 。在练习结束时，使用 PowerPoint 的小组要求使用 STP 来规划未来的练习。

过渡和部署

虽然美军在采用计算机技术方面极为保守，但是如今他们越来越清楚地明白，作战效率和培训正受到了用户界面各异，操作困难的各系统的阻碍。然而，这种认识需要时间才能传播到军队这种拥有超多军事和民间干系人的巨大组织，这些干系人包括作战用户和武器装备采办机构。除了技术开发之外，STP 开发团队用了多年的时间开展宣讲、演示、测试和相关活动来达到所需的关注程度，使其能够影响组织，让其采用 STP。在过去的那段时间里，虽然人们演示了 STP 的原型，但是他们仍然需要可以商用的语音识别来促成保守的决策者确定下列事实：在任务关键系统中纳入语音技术的风险已经大大降低。不仅如此，决策者自己独立地发现了界面的复杂性对他们组织中培训和作战的影响。然而，这种涉及组织变化的过程远未完成，因此客户的教育一直都会是个潜在的问题。现在，STP 已经被逐渐用于陆军情报实验分析部门（Army’s Intelligence Experimentation Analysis Element），陆军模拟和训练技术中心（Army Simulation and Training Technology Center）以及海军陆战队战争实验室的实验部门（Marine Corps Warfighting Laboratory’s Experiments Division），他们使用 STP 创建训练用的计划以及与仿真器进行集成。我们也看到陆军训练设施的人员表现了相当大的兴趣。相对于训练主题而言，设施中的工作人员花了太多的时间来训练学生使用 C2 系统。不仅如此，STP 不仅可以作为规划工具，人们对它的多通道技术也抱有相当大的兴趣，因为这种技术可用于车载计算机和手持设备上进行快速数据输入。

对于 STP 的完全部署，由议会制定的“项目记录（program of record）”采办流程规定了未来很多年的项目预算；新的技术现在很难被纳入这些项目，因为它们必须具有官方需要的能力且必须在竞争性功能排名流程中能被选中替换已有的预算项目。尽管存在这些障碍，STP 和多通道界面技术现在正在接受陆军项目行政办公室（Army’s Program Executive Office）的评估，以获取其与 C2 系统的集成情况。陆军项目行政办公室负责指挥和控制技术。

结语

我们已经说明了 STP 多通道界面处理现有 C2 GUI 面临的用户界面问题的方式。STP 学习和使用起来相当快，相当容易，而且支持很多不

h 方差检验使用的双样本 F 检验：$F(19) = 4.05$，$p < 0.02$

表 2STP 与实验对象之前使用的 C² 系统的对比方面的问卷调查结果

组织	系统与 STP 的对比	用户数量	STP 更易使用	STP 更快	STP 更好	更喜欢语音 / 草图
1 师	车载 C2 系统	41	83%	88%	81%	90%
2 师	车载 C2 系统	44	97%	97%	100%	87%
3 师	车载 C2 系统	37	78%	89%	84%	87%
	总体	122	87%	92%	89%	88%
2 师	CPOF	16	76%	94%	85%	88%
3 师	CPOF	12	88%	79%	84%	100%
4 师	CPOF	5	100%	100%	100%	100%
	总体	33	84%	89%	87%	94%

同的外形因子，包括手持设备，平板，车载，工作站和超移动的数字纸和笔。该界面利用和增强了士兵已经拥有的技能，因为他们已经接受过标准化语言、符号和军事决策过程方面的培训。凭借这种通用的“条令式”语言，STP 用户能够快速地创建行动方案或用多通道的方式输入 C2 和仿真系统中的作战数据，而不需要针对复杂的用户界面接受大量的培训。结果说明，可用性相当高的界面可以被集成到现有的 C2 系统中，进而在降低费用的同时提高用户的效率。

鸣谢

STP 的开发得到了小型企业创新研发计划（Small Business Innovation Research）第三期合同的资助，其中包括 DARPA 的 HR0011-11-C-0152，主合同 W15P7T-08-C-M011 下的 SAIC 分包合同，主合同 W15P7T-08-C-M002 下的 BAE Systems 分包合同，陆军研究、发展和工程指挥 / 仿真（Army Research, Development, and Engineering Command/Simulation）以及训练技术中心（Training Technology Center）的合同 W91CRB-10-C-0210。本文已经获批可以公开发表，其传播不受限制。本研究的结果和本文观点仅代表作者观点，不代表美国政府的观点。我们在此感谢 Todd Hughes，Colonels（退休）Joseph Moore，Pete Corpac 和 James Zanol 以及 ROTC（预备役军官训练团）的学生测试员。我们还要感谢 Paulo Barthelmess，Sumithra Bhakthavatsalam，John Dowding，Arden Gudger，David McGee，Moiz Nizamuddin，Michael Robin，Melissa Trapp-Petty 和 Jack Wozniak，他们在开发和测试 STP 时做出了相当大的贡献。同时感谢 haron Oviatt，General（退休）Peter Chiarelli，和多位匿名评审员。

参考资料

1. Alberts, D.S. and Hayes, R.E. *Understanding Command and Control*. DoD Command and Control Research Program Publication Series, Washington, D.C., 2006.
2. Bolt, R.A.Voice and gesture at the graphics interface. *ACM Computer Graphics 14*, 3 (1980), 262–270.
3. Cheyer, A. and Julia, L. Multimodal maps:An agent-based approach.In *Proceedings of the International Conference on Cooperative Multimodal Communication* (Eindhoven, the Netherlands, May).Springer, 1995, 103–113.
4. Clarkson, J.D. and Yi, J. LeatherNet:A synthetic forces tactical training system for the USMC commander. In *Proceedings of the Sixth Conference on Computer Generated Forces and Behavioral Representation Technical Report IST-TR-96-18*, University of Central Florida, Institute for Simulation and Training, Orlando, FL, 1996, 275–281.
5. Cohen, P.R., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L., and Clow, J. QuickSet:Multimodal interaction for distributed applications.In *Proceedings of the Fifth ACM International Conference on Multimedia* (Seattle, WA, Nov. 9–13).ACM Press, New York, 1997, 31–40.
6. Cohen, P.R. and McGee, D.R.Tangible multimodal interfaces for safety-critical applications.*Commun. ACM 47*, 1 (Jan. 2004), 41–46.
7. Cohen, P.R., McGee, D., Oviatt, S., Wu, L., Clow, J., King, R., Julier, S., and Rosenblum, L. Multimodal interaction for 2D and 3D environments.*IEEE Computer Graphics and Applications 19*, 4 (Apr. 1999), 10–13.
8. Courtemanche, A.J. and Ceranowicz, A. ModSAF development status.In *Proceedings of the Fifth Conference on Computer Generated Forces and Behavioral Representation*, University of Central Florida, Institute for Simulation and Training, Orlando, FL, 1995, 3–13.
9. Dowding, J., Gawron, J.M., Appelt, D., Bear, J., Cherny, L., Moore, R., and Moran, D. Gemini:A natural language system for spoken-language understanding. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics* (Ohio State University, Columbus, OH, June 22–26).Association for Computational Linguistics, Stroudsburg, PA, 1993, 54–61.
10. Dowding, J., Frank, J., Hockey, B.A., Jonsson, A., Aist, G., and Hieronymus, J. A spoken-dialogue interface to the EUROPA planner.In *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space* (Washington, D.C.).NASA, 2002.
11. Greene, H., Stotts, L., Patterson, R., and Greenburg, J. *Command Post of the Future:Successful Transition of a Science and Technology Initiative to a Program of Record*. Defense Acquisition University, Fort Belvoir, VA, Jan. 2010; http://www.dau.mil
12. Hammond, T., Logsdon, D., Peschel, J., Johnston, J., Taele, P., Wolin, A., and Paulson, B. A sketch-recognition interface that recognizes hundreds of shapes in course-of-action diagrams.In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems* (Atlanta, Apr. 10–15).ACM Press, New York, 2010, 4213–4218.
13. Hyman, P. Speech-to-speech translations stutter, but researchers see mellifluous future.*Commun.ACM 57*, 4 (Apr. 2014), 16–19.
14. Johnston, M., Cohen, P.R., McGee, D., Oviatt, S.L., Pittman, J.A., and Smith, I. Unification-based multimodal integration.In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Annual Meeting of the European ACL* (Madrid, Spain, July 7–12).Association for Computational Linguistics, Stroudsburg, PA, 1997, 281–288.
15. Johnston, M., Bangalore, S., Variredddy, G., Stent, A., Ehlen, P., Walker, M., Whittaker, S., and Maloor, P. MATCH:An architecture for multimodal dialogue systems.In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (Philadelphia, PA, July).Association for Computational Linguistics, Stroudsburg, PA, 2002, 376–383.
16. Kaiser, E., Olwal, A., McGee, D., Benko, H., Corradini, A., Li, X., Cohen, P.R., and Feiner, S. Mutual disambiguation of 3D multimodal interaction in augmented and virtual reality.In *Proceedings of the Seventh International Conference on Multimodal Interfaces* (Trento, Italy, Oct. 4–6).ACM Press, New York, 2005, 12–19.
17. Kara, L.B. and Stahovich, T.F.An image-based, trainable symbol recognizer for hand-drawn sketches. *Computers and Graphics 29*, 4 (2005), 501–517.
18. Kumar, S., Cohen, P.R., and Coulston, R. Multimodal interaction under exerted conditions in a natural field setting.In *Proceedings of the Sixth International Conference on Multimodal Interfaces* (State College, PA, Oct. 13–15).ACM Press, New York, 2004, 227–234.
19. MacEachren, A.M., Cai, G., Brewer, I., and Chen, J. Supporting map-based geo-collaboration through natural interfaces to large-screen display.*Cartographic Perspectives 54* (Spring 2006), 16–34.
20. Moran, D.B., Cheyer, A.J., Julia, L.E., Martin, D.L., and Park, S. Multimodal user interfaces in the Open Agent Architecture.In *Proceedings of the Second International Conference on Intelligent User Interfaces* (Orlando, FL, Jan. 6–9).ACM Press, New York, 1997, 61–68.
21. Myers, K., Kolojejchick, J., Angiolillo, C., Cummings, T., Garvey, T., Gervasio, M., Haines, W., Jones, C., Knittel, J., Morley, D., Ommert, W., and Potter, S. Learning by demonstration for military planning and decision making:A deployment story.In *Proceedings of the 23rd Innovative Applications of Artificial Intelligence Conference* (San Francisco, CA, Aug. 6–10).AAAI Press, Menlo Park, CA, 2011, 1597–1604.
22. O’Hara, K., Gonzalez, G., Sellen, A., Penney, G., Varnavas, A., Mentis, H., Criminisi, A., Corish, R., Rouncefield, M., Dastur, N., and Carrell, T. Touchless interaction in surgery.*Commun.ACM 57*, 1 (Jan. 2014), 70–77.
23. Oviatt, S.L.Multimodal interfaces.*The Human-Computer Interaction Handbook:Fundamentals, Evolving Technologies and Emerging Applications, Revised Third Edition*, J. Jacko, Ed. Lawrence Erlbaum Associates, Mahwah, NJ, 2012, 405–430.
24. Oviatt, S.L.Taming recognition errors with a multimodal architecture.*Commun.ACM 43*, 9 (Sept. 2000), 45–51.
25. Oviatt, S.L. and Cohen, P.R.Perceptual user interfaces:Multimodal interfaces that process what comes naturally.*Commun.ACM 43*, 3 (Mar. 2000), 45–53.
26. Oviatt, S.L. and Cohen, P.R.*The Paradigm Shift to Multimodality in Contemporary Computer Interfaces*. Morgan & Claypool Publishers, San Francisco, CA, 2015.
27. Stilman, B., Yakhnis, V., and Umanskiy, O. Strategies in large-scale problems.In *Adversarial Reasoning:Computational Approaches to Reading the Opponent’s Mind*, A. Kott and W. McEneaney, Eds. Chapman & Hall/CRC, London, U.K., 2007, 251–285.
28. U.S.Army.*U.S.Army Field Manual 101–5-1*, Chapter 5, 1997; http://armypubs.army.mil/doctrine/dr_pubs/dr_a/pdf/fm1_02c1.pdf

Philip R. Cohen（philcohen86@gmail.com）是 Adapx 的联合创始人，先进人工智能学会（Association for the Advancement of Artificial Intelligence）的研究员，曾任计算语言学协会的主席。

Edward C. Kaiser（ekaiser@sensoryinc.com）是俄勒冈州波特兰 Sensory 公司的高级应用工程师，曾任华盛顿州西雅图 Adapx 公司 STP 的项目共同负责人 / 项目负责人（2008–2009/2010）。

M.Cecelia Buchanan（mcbuchanan@gmail.com）是华盛顿州西雅图 Tuatara Consulting 的顾问，在编写本文时曾任华盛顿州西雅图 Adapx 的研究科学家。

Scott Lind（scott.lind@adapx.com）是华盛顿州西雅图 Adapx 国防和联邦解决方案部副总监。

Michael J. Corrigan（michael.corrigan@adapx.com）是华盛顿州西雅图 Adapx 公司的软件研究工程师。

R.Matthews Wesson（matt.wesson@adapx.com）是华盛顿州西雅图 Adapx 公司的高级研究程序员。

译文责任编辑：田丰

在本《通讯（Communications）独家视频中，您可以观看作者对本研究的讨论。

DOI:10.1145/2667218

植入性设备常常依赖软件，它们挽救了无数的生命。然而它们有多安全？

撰　稿　人：JOHANNES SAMETINGER、JERZY ROZENBLIT、ROMAN LYSECKY 和 PETER OTT

医疗设备的安全性挑战

医疗领域中的安全（safety）和安全保护（security）问题表现为许多不同的形式。例如，药剂恶意污染、血管支架召回和健康数据遭窃，等等。因无意威胁而造成的风险早已广为人知，如来自电磁能的干扰。

有意威胁带来的安全性风险近期才得以确认，因为医疗设备越来越多地使用更新颖的技术，如无线通信和互联网接入等。有意威胁包括未经授权访问医疗设备，或者未经许可更改此类设备的设置。人们常常引用美国食品和药品管理局 (FDA) 一位高级官员的话，“据我们所知，成百上千的医疗设备受到了恶意程序的感染。”[34] 虽然目前尚无此类入侵造成人员伤亡的报告，不难想象这样的一天终会到来。毋庸置疑，卫生保健将继续在数字化道路上前行。医疗设备也会越来越智能，越来越互联。这种趋势的一个副作用便是医院和诊所中的计算机面临着病毒风险。若无适当的对策，可能会导致更多的数据失窃，乃至危及患者生命的恶意攻击。

安全保护即保护信息和信息系统，防止其受到未经授权的访问和使用。如前文所述，越来越多的医疗设备中嵌入了具有通信机制的软件，它们已经可以归入信息系统的行列。信息的保密性、完整性和可用性是设计和运作上的核心目标。安全保护软件应当在遭受恶意攻击时也能发挥正确的作用。[25] 从这一角度而言，医疗设备安全性就是将这些设备设计为即使受到恶意攻击也能继续正常工作。这涵盖了内在的硬件和软件，以及外来的有意和无意威胁。

医疗设备包含多种多样的仪器和装置。在本文范围内，仅考虑带有硬件、软件和某种互操作能力的设备。例如，人工关节不会进行任何数据处理，也就是说，不涉及任何软件。所以，从安全保护 (security) 的角度而言我们可以忽略它们。但它们在安全性（safety）角度上或许非常重要。

» 重要见解

- **由于健康记录的敏感性、医疗设备互操作的增多，也因为人的幸福和生命处于风险之中，卫生保健领域呈现出安全性方面的挑战。**
- **植入性设备尤其是关键，因为它们倘若没有适当的保护就可能会让患者置于生命受到威胁的境地。**
- **全球数百万患者在使用医疗设备，其重要性不言而喻。它们越来越依赖软件，通过无线通信和互联网连接与其他设备进行互操作，这使得安全性成为其首要因素。**

图片由 MOPIC 提供

此时此刻，我们强调的是医疗设备安全保护的重要性。其实重点并不在于防止有人利用医疗设备杀害他人。不过，尽管这听起来遥不可及，这样的情景并非完全不可能。保护医疗设备安全是保护重要的基础设施。它在于防止恶意人员控制此类基础设施，防止勒索敲诈设备制造商或医疗保健机构，也关乎需要使用此类设备的人员的福祉。

动机

媒体几乎时常报导影响到公众的重大 IT 安全性事故，如密码失窃、信用卡信息盗用或网站宕机等。个人身份信息的丢失、遭窃或暴露这一重大问题也已在卫生保健领域中泛滥，占到了此类问题报告数量的两成。[33] FDA 正在收集关于医疗设备可报告问题的信息，以采集和确定某一或某种设备的不良和意外事件。每一年，数十万例医疗设备报告涉及了疑似与设备相关的死亡、重伤和故障。[6] 对此类召回和事件的分析表明，召回和不良事件的数字这些年已有增长。

设备召回的一大原因是故障。计算机相关的召回占了 20% 到 25%，这一数字还在攀升。数据表明计算机相关的召回主要由软件导致。[1] 90% 以上的设备召回在纠正措施的原因中提到了“软件”一词。不足 3% 提到可以在线获取升级。[23] 而且，Kramer 等人测试了 FDA 的不良事件报告机制；他们告知了某一设备的漏洞，但发现数月之后该事件才出现在对应的数据库中。对于回应软件相关的故障而言，这一时间显然太长。

多个案例中也显现出成功攻击医疗设备的可能性。例如，通过无线方式向胰岛素泵发送命令（提高或降低胰岛素水平，或者将它停用）。这可以在不超过 150 英尺的距离内完成。[20] FDA 在其“安全通讯 (Safety Communications)” 中向设备制造商和卫生保健提供商发布了一则警告，提醒他们实施安保举措来防止网络攻击。[9] 尽管尚无伤亡报告，但这样的后果显然可以想象。非医疗 IT 领域也可能会给医疗运作带来威胁。例如，如果某一防病毒程序将正常的系统文件识别为病毒，造成全球许多计算机停机，医院将不得不推迟可择期的手术并且停止治疗患者。[11]

医疗设备

医疗设备包含从简单的木质压舌板和听诊器到非常精密的计算机化医疗设备等一切事物。[37] 根据世界卫生组织 (WHO) 的定义，医疗设备是“一种仪器、器具、装置、机器、体外试剂，或者其他类似或相关的物体”，用于疾病或其他状况的诊断、预防、监控和治疗等。[37] FDA 使用了类似的定义。[7] 医疗设备的分类在美国、加拿大、欧洲或澳大利亚等不同国家 / 地区各不相同。FDA 为大约 1,700 种不同的通用设备指定了分类。这些设备分入称为属组 (panel) 的医疗专科中。例如，FDA 的专科属组包含心血管设备、牙科、整形科，以及耳鼻喉器械等。主动型设备可能涉及也可能不涉及软件、硬件和接口，而考虑安全性问题时这些很重要。这些设备可以执行一些数据处理，从外部（传感器）接收输入，向外界（传动装置）输出值，并且与其他设备通信。

设备安全。每一种 FDA 的通用设备类型分配到三个管理类别之一：I、II 和 III。这些类别基于确保设备安全和效用所需的控制级别；风险越高，类别号越大。[8] 例如，III 类设备必须通过上市前审批流程的批准。这一类别中包含永久植入人体、可能作为生命延续基础的装置，如人工心脏或自动体外除颤器。这种分类基于设备对患者或用户造成的风险。I 类包含风险最低的设备，III 类包含风险最高的设备。

根据 WHO 的设想，最佳的医疗设备安全和性能需要该设备生命周期中涉及的各个方面（即政府、制造商、进口商 / 出口商、用户和公众）之间协作进行风险管理。[37] 国际标准 ISO 14971:2007 为医疗设备制造商提供了一个框架，其包括设备设计、开发、制造，以及设备安全和性能售后监控过程中风险管理的风险分析、风险评估和风险控制。[18]

设备安全保护性。我们认为，在安全保护性上重要的医疗设备通常会执行某种形式的数据处理和通信，常常通过在专用硬件上运行某种形式的软件，而且时常部署有一系列的传感器。[7] 传感设备构成了安全性威胁，因为传感器数值出错可能会造成医生或医疗设备做出错误的治疗决定。安全关键信息影响着个人或环境的安全。例如，植入式除颤器或 X 光机的参数设置或命令。此类信息的恶意和无意修改或可导致重大安全事故。敏感信息包括关于患者的任何信息，如医疗记录；它也包括传感设备的数值，其报告了关于个人或设备状态的信息，如血糖水平、ID 或起搏器参数设置。需要注意的是，根据 WHO 或 FDA 的定义，所有医疗设备都与生俱有关乎安全的层面。一些风险较高，一些风险较低（见 FDA 的类别 I、II 和 III）。不过，并非所有这些设备都与安全保护性相关；如上文中提到的人工关节。通常而言，一旦牵涉到软件，就会有安全保护性问题。不过，还有一些安全保护性相关的设备并不被 WHO 或 FDA 视为医疗设备。例如，运行处理敏感信息的医疗应用程序的智能手机，或者医院中处理医疗记录的普通电脑。

安全性和安全保护性之间的区别并不始终明显，因为安全保护性对安全有显著的影响。通常而言，安全性在于保护设备的环境（主要是患者）免受设备本身的危害。制造商必须确保设备不会伤害患者；例如，不在植入体中使用有毒物质，或者谨慎开发胰岛素泵的软件。安全保护性在于保护设备免受其环境的影响，正好与安全相对。只要设备使用独立的模式运作，这就不是问题。然而，如果设备与其环境通信或与互联网或其他系统连接，就会有人能够访问设备上的数据，甚至加以控制。当恶意攻击者获得设备的控制权并且伤害到患者时，安全保护性问题就演变为安全性问题。

如果攻击者在设备安装之前成功植入了恶意硬件或软件，无通信功能但有处理能力的设备也就有了安全保护性上的问题。例如，将硬件或软件木马安装在心脏起搏器中，并在特定事件时激活。设计和开发过程中必须采取预防措施，从而避免此类攻击。显然，具备通信功能的必须提供了更宽的“攻击面”。

我们建议，根据是否处理或通信敏感信息、是否处理或通信安全攸关的信息，对医疗设备进行安全性方面的分类。附表中总结了我们为安全性相关设备建议的分类级别。请注意这一组合为初始分类。尽管还没有完整地阐述，它是制定更为全面的安全性级别分类的第一步。

卫生保健专业人士通过移动医疗应用程序改善和促进患者护理。越来越多的患者通过此类应用程序管理其健康和保健。此类应用程序可以提升健康生活水平，并且提供对实用健康信息的访问。移动医

疗应用程序也有多种多样的用途。它们通过和医疗设备连接来提供扩展，显示、存储、分析或传输患者相关的数据。并非所有移动医疗应用程序都带来安全性风险。然而，只要它处理或传输敏感信息，甚至控制医疗设备，那就必须在安全保护性上采取预防措施。

起搏器场景

我们以起搏器为例来阐述安全保护性问题。起搏器是植入患者体内以调节患者心率的医疗设备。此类设备的用途是让患者保持适当的心率；若无它的帮助，患者的心脏可能无法正常跳动。起搏器归类为 III 类，即安全要求最高的类别。

临床角度。植入性医疗设备遍布在许多医疗专科中。植入性心脏起搏器和除颤器对于患者的健康和福祉极为重要。此类设备每年植入到数十万患者的体内；如果没有功能完全正常的设备，这些患者中有许多将无法生活。植入了此类设备的患者定期参加随访，前往诊所或医院对该设备进行检查和必要的调整。经过训练的职员或医师利用供应商的程序系统执行这样的功能，这些系统通过扫描或无线技术与设备进行通信。此外，近几年中几乎所有设备供应商都建立了一套家用的设备随访系统。为此，数据模块配置到患者家中（通常是床边）。患者接近该数据模块时，无线通信便会建立，数据模块就能检查该设备。此信息（通常通过电话线）发送到基于互联网的存储库。经过授权的卫生保健专业人士可以查看此信息。

植入性心脏起搏器和除颤器具有很高的可靠性。不过，设备组件故障也有发生，凸显了潜在的医疗和法律隐患。这些故障大部分源于制造工艺和 / 或材料的问题，而且通常限于特定的设备批号。然而，此类设备故障几乎都需要通过手术来更换设备。随着基于网络的远程设备随访系统的普及，对于设备安全性的担忧也日渐加重。目前，这样的远程随访系统处于只读模式。但通过远程随访系统进行设备编程正在研究之中。由于失误、技术故障或恶意目的造成的编程错误可能给患者带来危及生命的风险。

风险评估。在我们的起搏器场景中，我们根据 CIA 三部曲、保密性、完整性和可用性来区分不同的风险。首先是保密性，有关患者和起搏器的敏感数据可能会被泄露。其次，完整性；设备上的数据可能会被篡改，为患者带来或轻或重的各种影响。第三，可用性；可能表现为设备不能运作。第 79 页的附图中提供了起搏器环境的结构概览。虽然起搏器本身以无线方式通信，其他通信则可能通过互联网、电话线进行，有时甚至还利用优盘。即便编程设备目前还未直接与诊所相连，但将来迟早如此。

信息泄露和篡改可以在设备之间的任何连接上发生。互联网上可能会发生中间人攻击，除非采取了加密机制等适当的措施。此外，无线通信也使得攻击者能够借助另一设备侦听流量，如另一编程设备、另一家用监控器，或者专用于攻击的其他设备。此类设备不仅可用于侦听，而且还能伪装成授权的通信方。拒绝服务攻击也会发生。在我们的场景中，最大的威胁来自起搏器的互操作性。设备安全风险评估的目的是判断风险的性质、其危害程度，以及危害的发生几率。[27] 必须根据这一信息来确定和选择应对措施。

软件安全保护性。软件漏洞是软件中的错误和纰漏，可直接被攻击者用于获取系统或网络的访问权限。起搏器的软件具有保密性和专有性。有一个面向学术用途开放的起搏器系统规格说明。[2] 展示了这些看似简单的设备的复杂性。里面有许多可以编程的参数；例如，调高和调低心率限值，以及各种各样的延时和周期。功能则包括设备监控、引线支持、脉搏起搏、各种运行模式和状态，以及广泛的诊断功能。不仅起搏器本身需要软件，编程设备和家用监控器上也有需求。编程设备上的软件用于非侵入地重新编程起搏器；例如，修改起搏器频率、监控特定的功能，以及处理从起搏器获取的数据。此类软件可被用于一种或多种型号的设备，它们通常来自同一制造商。家用监控器上的软件必须与起搏器通信，主要用于将重要的信息上传到特定的服务器，供诊所的人员访问。编程设备和家用监控器，甚至起搏器本身可能需要安装软件更新。被攻破的起搏器可能会直接危害其患者。被攻破的编程设备可以间接地产生危害。或许仅仅是将心脏病科医师选定之外的其他参数发送至设备。被攻破的家用监控器也可带来严重的威胁。如果它将错误的数值发送

医疗设备的安全保护级别

安全保护级别	说明	设备示例
低	没有敏感或安全攸关的活动	医院中用于行政工作的电脑心率监测手表
中	敏感活动	处理健康电子记录 (EHR) 的电脑传输血糖水平的智能手机
高	安全攸关的活动	控制胰岛素泵或发送参数至起搏器的设备
很高	安全攸关的活动，从别处获取输入	接收外部参数的起搏器

至服务器，这些数值就可能会导致心脏病科医师做出错误的结论，最终导致设备设置错误而可能危害患者。最后但并非最不重要，存储这些数值的服务器失守时也会产生类似的威胁。

硬件。隐藏的恶意电路为攻击者提供隐蔽攻击的途径。[21] 权限提升、登录后门和密码盗窃等各种潜在攻击已被展示过。和软件一样，起搏器的硬件也具有保密性和专有性。明尼苏达大学提供了硬件参考平台。它以 8 位微控制器为基础。[26] 编程设备和家用监控器的硬件受到的制约较少一些。

这些设备没有空间和功率限制，和一般的电脑差不多。与软件相似，恶意硬件电路可以被放入医疗设备本身，也可置入与其通信的其他设备，如我们起搏器例子中的编程设备和家用监控器。存储起搏器数据的网络服务器上的恶意硬件也会带来威胁，如泄露敏感的医疗数据，或者篡改此类数据进而误导提供治疗的医师。

互操作。起搏器安全保护性问题的提出也源自其无线通信能力。相关的担忧包括未经许可访问设备上的患者数据，以及未经授权改动设备的参数。

毋庸置疑，篡改设置可能危及害患者，给其心脏造成严重危害，甚至导致其死亡。设备的无线通信受到攻击时，其完整性就岌岌可危。关键问题在于未经授权的第三方是否有可能更改设备设置，更改或停用疗程，或者甚至发起命令冲击。Halperin 等人借助示波器和软件无线电对起搏器的通信协议进行了部分反向工程，实施了一些能够危及患者安全和隐私的攻击。[15]

即使我们起搏器例子中的所有设备的都没有恶意的硬件和软件，攻击者依然能够通过与这些设备中任何一个通信来造成威胁，如家用监控器、编程设备、服务提供商网络服务器，或者起搏器本身。互操作需要协议来定义两个通信参与方之间的操作序列。这些序列必须确保数据保护。网络协议常常遭受漏洞的困扰，使得攻击者能够将自身伪装成他人。攻击者可能会利用篡改过的、天线更强的编程设备，从更远的距离与起搏器通信。然后，他们伪装成已获授权的心脏科医师，修改设备上的设置。类似地，他们也许能冒充家用监控器来读取敏感数据，或者冒充起搏器与家用监控器通信来传输错误的数值。

医疗设备安全性就是将这些设备设计为即使受到恶意攻击也能继续正常工作。

挑战

卫生保健领域中值得妥善保护的关键资产包括医疗记录、许许多多的医疗传感器和医疗设备，以及最终（但并非最不重要）的人类健康和生命。与普通的 IT 安全保护性相比，医疗设备的安全保护性不仅有所差别，而且更具挑战性；其原因有许多，不仅仅是因为人的生命处于风险之中。当然，汽车等非医疗设备在由于安全保护性漏洞而丧失安全性时，也会危及人的生命。我们可以想象这样的情景，恶意程序植入到动力稳定控制系统中以故意造成事故。但许多医疗设备影响着患者的生理机能，因而会带来永久的威胁。虽说不是全部，但许多重要的植入性医疗设备存在资源限制。稀少的内存、处理能力，以及物理尺寸局限和电池续航时间限制了安全保护性对策的用武之地。紧急情况带来其他领域中没有的额外挑战。医疗设备必须防止未经授权的访问，但又需要允许在紧急情况时进行简单而快速的访问。另一问题是再现性。安全防护性研究人员常常缺少对专有设备的访问权限，因而在研究攻击和防御的能力上受到束缚。

相关论文介绍了多种医疗设备的漏洞对策。[4,14] 它们可能具有防护、

纠正或检测的特质。例如，审计、通知、可信外部或内部设备，以及加密保护等。[16] 本文中，我们列举了各种挑战，也设想了应对它们的举措。

软件安全保护性。除了功能外，医疗设备的软件开发者还必须采取措施，为其代码的安全性和安全保护性提供保障。安全开发和安全更新机制都不能少。Fu 和 Blum 的论文中描述了医疗设备软件的风险。[12]

*安全开发。*安全保护性是一种反复无常的属性。系统永远不能百分百安全。只要漏洞不为人所知，它就不是问题。当攻击者知道漏洞具体为何时，目标系统就处于风险之中。安全医疗软件的开发与其他类型的软件没有本质上的差异。人们有一种常识性错误，只有低能的程序员才会写出不安全的代码。除了代码编写的复杂性，还需要详尽的知识、附加的培训，以及额外的开发活动才能编写出安全的代码。[17] 因此，经济乃至社会方面的因素常常会拖累安全性质量。

在医疗设备软件中，我们必须确保安全性与安全保护被视为头等大事，并且设有既定的流程来报告和修复漏洞。医疗设备还有其他的挑战，用于安全性的额外代码不得妨碍实时限制以及电池电量有限等资源约束。

*更新机制。*当系统的制造商获悉漏洞时，他们将应对并且纠正问题。然后必须将补丁分发到具有该漏洞的系统。更新机制本身可能会被滥用于攻击。与个人电脑和智能手机相比，医疗设备的更新和补丁程序的频率（仍然）要低得多。然而，有时候它们是不可或缺的。

我们需要对医疗设备采用用户友好的更新流程，并且要采取预防措施确保更新流程本身不会沾染恶意程序。此外，更新不得打破或中断设备的正常运作。

*现成的软件*通常被用作医疗技术的“支撑”。在医疗设备上，软件补丁或更新常常会被延迟或完全缺失。缺少补丁也可能是组织流程上的问题。延迟或许来源于设备制造商必须批准软件升级，以及任何安全性安装。[36] 旧版软件的问题在于它们包含已知的漏洞。

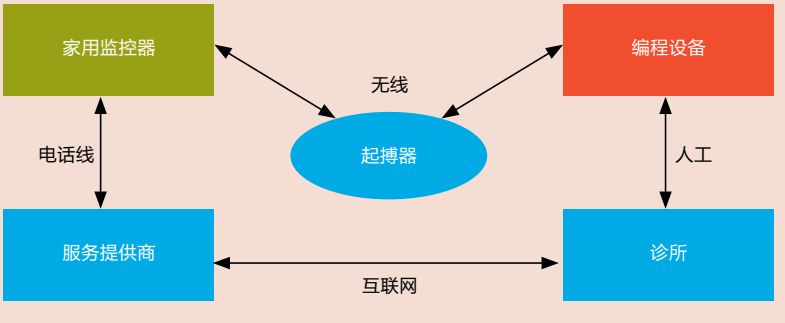
只要医疗设备独立运作，其包含的旧软件就不会是问题。互联的增加使得这些设备也能受到旧的恶意程序的攻击。[12] 对于医疗设备而言，嵌入式软件的产品生命周期必须与该设备的产品生命周期相匹配，这一点很重要。制造商必须确保医疗设备不会用到支持周期已过期的软件。

硬件安全性。在硬件上，安全问题要比安全保护问题更加普遍。例如，非医疗设备对起搏器的电磁干扰。医疗设备中的硬件木马目前似乎不太现实，但必须尽可能采取预防措施来减少攻击途径。军用芯片中存在后门已有记载；攻击者可以从芯片提取配置数据，重新编制密码和访问密钥，修改低级芯片功能，以及永久性地损坏设备。[30] 相关研究也展示了将可定制硬件木马自动嵌入到任意有限状态机中的方法。这些木马无法检测或修复。[35] 电脑中也曾被嵌入过无线电传播途径，使得它们即使没有连接互联网也能被远程控制和注入恶意程序。[29]

我们必须要谨记，硬件木马也能够成为医疗设备的攻击途径。务必要确保制造流程不会被此类恶意程序钻到空子。由于对电脑辅助设计工具的依赖，也需要进一步确保这些工具不会在设计过程中插入硬件木马。硬件设计过程中要运用验证方法，确保设计的输出与输入相符，不会包含额外的电路。除了在设计的各个阶段使用信赖的产品外，别无他法，确保硬件毫无木马不切实际。因此，仍然需要检测和消除功能。一旦检测到恶意硬件并且清楚其行为，对于如何消除恶意硬件的影响以确保医疗设备的安全的研究具有至关重要的意义。

互操作。医疗设备对无线功能的依赖度越来越高，无论是用于远程监控，还是远程更新设置、甚至软件本身。互操作方面的挑战包括安全协议、身份鉴定、授权、加密和密钥管理。医疗紧急状况使得医疗设备的互操作性变得尤为棘手。在可能会危及生命的紧急状况中，医疗人员可能不仅需要访问医疗记录，同时也需要访问患者的医疗设备。对于此类状况，鉴定和授权机制必须拥有旁路或捷径。然而，这些旁路或捷径不能成为攻击者获取设备访问权限的手段。

起搏器环境。



保护医疗设备互操作的举措涵盖了体外佩戴的装置，[3] 如可信赖的手环护身符，[31] 以及软件无线电屏蔽等。[13] 研究人员已经创造了一种防火墙原型，它可以防止黑客干扰无线医疗设备，[32] 并且利用物理接触和 ECG 读数对比来验证身份。[28]

组织层面。从最初的开发周期开始就在系统中加入安全防护性设计，是最有效用的方法。务必要开发和维护威胁模型，并在设备开发期间评估风险。也需要制定提供软件更新和补丁的系统性计划。最后但并非最不重要，必须有安全响应团队持续地识别、监控和解决安全性的事件和漏洞。

为此，应当要鼓励医院和诊所等用户设施报告安全性事件。这些报告可以为医疗设备的安全性问题提供宝贵的见解。此外，我们建议为医疗设备确定安全性与威胁等级，为涉及的所有各方制定行动规则和审计准则。本文附表中定义的等级仅仅是这一方向迈出的一小步。我们希望通过简单的评分来总结医疗设备的敏感度、影响度、暴露程度，以及当前的威胁水平。然后，通过基于规则的行为引发必要的行动，对安全性相关的事件作出回应。

管理。在任何时候，务必都要清楚危险的级别并采取适当的对策。医疗设备的设计与分发受到严密的管控。在美国，FDA 是管理医疗设备分发的权威机构。制造商对医疗设备已获批准的配置负有责任。医院和患者等医疗设备用户无权访问其软件环境，也无法安装额外的安全性措施。任何升级或更新 — 无论是功能增加或安全性措施 — 通常都需要得到制造商的批准。因此，安全性相关升级的部署常常会被延后。[36] 制造商、进口商和设备用户设施必须报告具体的设备相关不良事件和产品问题。

必须重新思考监督策略，才能高效、有效地收集医疗设备中安全性与隐私问题的相关数据。[23] Fu 和 Blum 等人的论文对一些管理方面以及标准团体、制造商和临床机构的角色进行了探讨。[12] 我们看到了行动需求，必须针对医疗设备软件更新需求的增长以及重大变化后重新进行临床试验的需求而做出调整。

恶意程序检测。在检测到利用漏洞的恶意程序之前，相关的漏洞通常都处于未知状态。我们需要检测恶意程序的方法。恶意程序检测技术包含控制流完整性验证、调用堆栈监控、数据流分析，以及基于哈希的多源验证。虽然基于软件的恶意程序检测方式适合传统的计算机系统，具有严格时间限制的医疗设备可能无力承受其性能开销。基于硬件的检测方式可以减少或消除性能开销，但功耗依然是难点。

对于医疗设备，我们需要功耗非常低的非侵入式恶意程序检测方式，因为功率是宝贵的资源，尤其对植入性设备而言。为提供对零日攻击的抵御能力，需要基于异常的恶意程序检测方式。这些方式依赖正常系统行为的准确模型，这就需要对此行为进行建模的形式化方式，以及和系统设计任务的紧密整合。医疗设备中计时要求的重要性或许提供了一个独特的系统特征，可被用来更好地检测恶意程序。

恶意程序响应。检测恶意程序仅解决了一半问题。检测到恶意程序时，医疗设备当如何响应？通知是直接明了的选择，但会让恶意程序继续保持活动状态，直到设备被检查或更换为止。如果暂停设备对患者而言是安全的，或许可以重新安装软件。我们生活在一个互联互通的世界中。脱离互联网也许会给人带来一点失落，但不会造成生理上的伤害。然而，生命攸关的医疗设备呈现出一组更加复杂的挑战。对于按需起搏器这种只有心律异常时才起搏心脏的设备而言，任何重新编程、复位或连接中断的破坏性显然要小于对永久性起搏器执行这些操作的时候。这样的情形中必须考虑折衷方案。更换设备或许是个选择，但设备被更换之前的这段时间呢？能够关闭与设备的任何通信至少是首要的一步，美国前任副总统正是采取这个步骤避免了一场可能的恐怖袭击。[22] 但必须要明确的是，如果恶意程序已被植入在设备上，终止通信功能或许已经为时过晚。这种情形中，设备复位也许是个选择。

通知可以提醒患者潜在的恶意活动。[15] 然而，对已经忧心忡忡的患者而言，通知安全保护性漏洞可能雪上加霜。我们设想的是，或许可以在怀疑或者确认恶意程序时切换到不同的设备模式。例如，有那么一种模式能够关闭通信功能，并且使用预定义的安全参数设置。从挑剔的角度看，趋于改变的安全模式的设计必须确保各种软件实施得到隔离，无论是借助软件防护还是安全硬件架构，这样恶意程序就无法改变安全模式的运行。失效保护功能必须保护设备的关键功能，即使在安全保护性失守的时候。[10]

形式化方法。在部署到医疗设备之前查找软件和硬件中的漏洞可以大大提高安全性。在现实中，消除所有安全保护性漏洞是不可实行、不切实际的。可以运用形式化验证方法来分析当时的行为，检测潜在的漏洞。[24] 在开发心脏起搏器等安全关键型实时系统时，保障时序属性是重要的事项。Jee 等人以起搏器作为案例研究，展示了一种可确保安全的实时软件开发方法。[19] 他们遵循模型驱动的开发技巧，并采用基于测量的时序分析，从而在他们的实施和形式模型中保证时序属性。

形式化方法在确保医疗设备的硬件和软件按照设计正常运行方面发挥重要的作用。我们也认为，应当利用形式化方法来验证软件更新、恶意程序响应方法和其他运行时系统重新配置的正确性。确保运行时对系统的更改能够在不影响设备行为的前提下完成，形式化建模和验证是关键。

资源限制。有限的功率 / 能量和局促的尺寸或许让已知的安全性解决方案变得不切实际。例如，植入性除颤器可能没有运行商用防病毒软件的资源。即使它能够运行，也可能会消耗太多电池电量。此外，此类软件也必须连接互联网使病毒信息保持最新，因而也就开放了另一条攻击途径。有限的内存或许导致必须使用操作系统的缩减版本。它也使得对常见安全软件的利用变得难上加难。[36]

近期的研究表明，微小的发电装置可以将心脏跳动的动能转换为电能，而植入性设备也可通过无线方式进行充电。利用 RF 感应能量、不消耗电池电量的零功率通知和身份鉴定装置也已出现；例如，可用于通过声音警告患者安全性敏感的活动。[15] 但有限的资源仍然会束缚许多医疗设备中的安全保护性举措。

非技术层面。除了医疗设备的技术安全性层面外，我们也必须考虑非技术的问题。安全保护性意识是一个主要方面。例如，如果将登录凭据提供给未经授权的人，技术上的安全保护性措施就形同虚设。技术上可行的系统或许并非患者想要的。

普罗大众越来越担忧日常生活中许多方面对互联网的滥用，如银行欺诈或身份盗窃。作为心脏病学家和电生理学家，本文的一名作者 (P. Ott, M.D.) 发现患者对安全保护性问题的意识已有加强，他们质疑植入式设备在数字领域中的安全。

攻击者可能会利用篡改过的、天线更强的编程设备，从更远的距离与起搏器通信。

我们相信这样的担忧会带来越来越大的压力。一项小规模研究表明，在为医疗设备设计安全保护性对策时必须考虑感知上的安全性和安全安全保护性、自我形象，不会产生不必要的文化和历史方面的联想。[5]

我们需要更多信息了解患者对他们所用设备的安全性的担忧程度。用户研究或许显露出患者愿意采取哪些具体的额外步骤来提高安全性。这将给制造商提供宝贵的信息。我们需要提高制造商、患者、医生和医疗机构等所有相关方的安全保护性意识。此外，设备的安全保护性性状态必须更加醒目、易懂，能够为所有相关方访问。

IT 基础设施。为保护医疗设备，周遭的 IT 环境也必须得到妥善防护。由于本文关注的是医疗设备，我们不再赘述 IT 安全保护性中可见的一般对策。这些也适用于卫生保健安全保护性或医疗设备安全保护性；例如，处置硬盘之前擦除其数据、备份数据或 BYOD（自带设备）政策。智能手机或平板电脑等现成设备也越来越多地存储、处理和传输敏感的医疗数据。这些数据必须妥善保护，防止受到此类设备上恶意程序的侵害。

根据（美国）健康保险携带和责任法案 (HIPAA) 的规定，IT 基础设施必须保证医疗数据的隐私。然而，安全也在风险之中。对于医疗设备，务必要铭记普通的 IT 设备在直接或间接地和医疗设备互操作时也会给后者带来威胁。最为重要的是，医疗设备应当始终将其周围环境有可能失守作为前提。

总结

保护医疗设备意味着保护人的生命、健康和幸福。它关乎着敏感健康信息的隐私保护。我们发现对移动医疗应用程序的使用在增长，使用无线通信和互联网连接的医疗设备也在变多。新型传感技术为远距

离医疗带来机遇，卫生保健有望变得更加经济实惠。若不采取适当的对策，我们就会为敏感医疗数据的滥用、甚至危及人的生命的恶意程序和攻击敞开大门。 C

参考资料

1. Alemzadeh, H., Iyer, R.K. and Kalbarczyk, Z. Analysis of safety-critical computer failures in medical devices. *IEEE Security & Privacy 11*, 4, (July-Aug. 2013), 14-26.
2. *Boston Scientific*.PACEMAKER System Specification.2007.
3. Denning, T., Fu, K. and Kohno, T. Absence makes the heart grow fonder:New directions for implantable medical device security.In *Proceedings of USENIX Workshop on Hot Topics in Security*, July 2008.
4. Denning, T., Matsuoka, Y. and Kohno, T. Neurosecurity:Security and privacy for neural devices. *Neurosurgical Focus 27*, 1 (July 2009).
5. Denning, T. et al.Patients, pacemakers, and implantable defibrillators:Human values and security for wireless implantable medical devices.In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, 2010.
6. Food and Drug Administration.MAUDE—Manufacturer and User Facility Device Experience; http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfMAUDE/search.CFM
7. Food and Drug Administration.Is The Product A Medical Device? http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/Overview/ClassifyYourDevice/ucm051512.htm
8. Food and Drug Administration.Medical Devices – Classify Your Medical Device; http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/Overview/ClassifyYourDevice/default.htm
9. Food and Drug Administration Safety Communication:Cybersecurity for Medical Devices and Hospital Networks; June 2013. http://www.fda.gov/MedicalDevices/Safety/AlertsandNotices/ucm356423.htm
10. Food and Drug Administration.Content of premarket submissions for management of cybersecurity in medical devices—Draft guidance for industry and Food and Drug administration staff, June 14, 2013; http://www.fda.gov/medicalDevices/Deviceregulationandguidance/guidanceDocuments/ucm356186.htm
11. Fox News.Antivirus Program Goes Berserk, Freezes PCs.Apr. 22, 2010.
12. Fu, K. and Blum, J. Controlling for cybersecurity risks of medical device software.*Commun.ACM 56*, 10 (Oct. 2013), 35–37.
13. Gollakota, S. et al.They can hear your heartbeats:Non-invasive security for implantable medical devices. In *Proceedings from SIGCOMM'11* (Toronto, Ontario, Canada, Aug. 15–19, 2011).
14. Halperin, D. et al.Security and privacy for implantable medical devices.*IEEE Pervasive Computing, Special Issue on Implantable Electronics*, (Jan. 2008).
15. Halperin, D. et al.Pacemakers and implantable cardiac defibrillators:Software radio attacks and zero-power defenses.In *Proceedings of the IEEE Symposium on Security and Privacy*, May 2008.
16. Hansen, J.A. and Hansen, N.M.A taxonomy of vulnerabilities in implantable medical devices.In *Proceedings of SPIMACS'10*, (Chicago, IL, Oct. 8, 2010).
17. Howard, M. and Lipner, S. *The Security Development Lifecycle*.Microsoft Press, 2006.
18. International Standards Organization.Medical devices—Application of risk management to medical devices.ISO 14971:2007.
19. Jee, E. et al.A safety-assured development approach for real-time software, Proc.IEEE Int.Conf.Embed. *Real-time Comput.Syst.Appl.* (Aug. 2010), 133–142.
20. Kaplan, D. Black Hat:Insulin pumps can be hacked.*SC Magazine*, (Aug. 04, 2011).
21. King, S.T. et al.Designing and implementing malicious hardware.In *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. Fabian Monrose, ed. USENIX Association, Berkeley, CA.
22. Kolata, G. Of fact, fiction and Cheney's defibrillator. *New York Times*, (Oct. 27, 2013).
23. Kramer, D.B. et al.Security and privacy qualities of medical devices:An analysis of fda postmarket surveillance. *PLoS ONE 7*, 7 (2012), e40200; doi:10.1371/journal.pone.0040200
24. Li, C., Raghunathan, A. and Jha, N.K.Improving the trustworthiness of medical device software with formal verification methods.*IEEE Embedded Systems Letters 5*, 3 (Sept. 2013), 50–53.
25. McGraw, G. Software security.*IEEE Security & Privacy 2*, 2 (Mar-Apr 2004), 80–83.
26. Nixon, C. et al.Academic Dual Chamber Pacemaker. University of Minnesota, 2008.
27. Ross, R.S.Guide for Conducting Risk Assessments. NIST Special Publication 800-30 Rev. 1, Sept. 2012.
28. Rostami, M., Juels, A. and Koushanfar F. Heart-to-Heart (H2H):Authentication for implanted medical devices.In *Proceedings for ACM SIGSAC Conference on Computer & Communications Security*.ACM, New York, NY, 1099–1112.
29. Sanger, D.E. and Shanker, T. N.S.A. devises radio pathway into computers.*New York Times* (Jan. 14, 2014).
30. Skorobogatov, S. and Woods, C. Breakthrough silicon scanning discovers backdoor in military chip, cryptographic hardware and embedded systems. *Lecture Notes in Computer Science 7428* (2012), 23–40.
31. Sorber, J. et al.An amulet for trustworthy wearable mHealth.In *Proceedings of the 12th Workshop on Mobile Computing Systems & Applications*.ACM, New York, NY.
32. Venere, E. New firewall to safeguard against medical-device hacking.Purdue University News Service, Apr. 12, 2012.
33. Vockley, M. Safe and Secure?Healthcare in the cyberworld.AAMI (Advancing Safety in Medical Technology) BI&T – Biomedical Instrumentation & Technology, May/June 2012.
34. Weaver, C. Patients put at risk by computer viruses. *Wall Street Journal* (June 13, 2013).
35. Wei, S., Potkonjak, M. The undetectable and unprovable hardware Trojan horse.In *Proceedings of the ACM Design Automation Conference* (Austin, TX, May 29 – June 07, 2013).
36. Wirth, A. Cybercrimes pose growing threat to medical devices.*Biomed Instrum Technol.45*, 1 (Jan/Feb 2011), 26–34.
37. World Health Organization.Medical device regulations:Global overview and guiding principles.2003.

Johannes Sametinger (johannes.sametinger@jku.at) 就职于奥地利林兹的约翰开普勒大学，是该校信息系统系副教授。

Jerzy Rozenblit (jr@ece.arizona.edu) 就职于美国亚利桑那州图森的亚利桑那大学，是该校电气与计算机工程系 / 外科系特聘教授。

Roman Lysecky (rlysecky@ece.arizona.edu) 就职于美国亚利桑那州图森的亚利桑那大学，是该校电气与计算机工程系副教授。

Peter Ott (ottp@email.arizona.edu) 就职于美国亚利桑那州图森的亚利桑那大学，是该校萨维尔心脏中心医学院副教授。

译文责任编辑：陈文光

DOI:10.1145/2735839

如要查看随附论文
请访问 doi.acm.org/10.1145/2735841

技术视角
计算机硬件的专用化趋势

作者：Trevor Mudge

随着技术节点逐步缩小，登纳德缩放比例定律将不再适用，这已经成为计算机架构设计师面临的一个新的挑战。过去，根据登纳德缩放比例定律，芯片的运行速度上升，功率就会下降，面积也会缩小。并且，根据摩尔定律，这三者的同时上升不会带来芯片成本的明显增加。由于基于 CMOS 的技术的出现，摩尔定律也将寿终正寝。而且，随着工艺的特征尺寸过渡到 20nm 以下，至少其中两项 - 速度和功率 - 会逐渐停止改善。事实上，一些专家们认为，在单个晶体管的成本方面，28nm 工艺是最便宜的。

缩放的放缓将对电脑运算所涉各领域产生深远的影响。随着更多功能、更严功率限制在未来将成为常态，硬件系统的设计方式也将首先受到影响。这一点在移动平台领域最为明显，尤其智能手机，不断追求更多功能、更长电池使用时间。要满足更为严格的性能功耗比要求，仅靠过渡到下一代通用计算机是远远不够的。

专用化是提高能源效率的一个替代方案，但只有在有显著需求的情况下，该方案才能产生经济价值。要想找到平衡点，可以设计适用于特定应用领域、具有一定程度可编程性并具有足够应用空间的部件。本论文所介绍的研究做到了这一点。

作者介绍了一个可编程卷积引擎，它采用一个 1D 或 2D 模板，并与一个 1D 或 2D 数组（2D 图像便是一个典型例子）相卷积。模板比图像小很多。模板大小可低至 3x3 像素，而图像则可高达数亿像素。

作者详细介绍了他们的研究，以证明他们的设计在哪些方面填补了固定功能硬件和通用硬件之间的空白。

模板的大小是可编程的，这正是它的功能所在。该功能是可以广泛适用的，特别是在图像处理、计算机视觉以及虚拟现实这一新兴领域。此类功能已在各种移动平台上快速运用，证明移动平台需要高能效地使用此类功能，因为调整专门的卷积处理器并不难。这一观点是促使作者研究可编程卷积引擎的动机。

就可编程卷积引擎本身而言，还需做出重要的设计选择，才能使绝大多数决策实现专用化。为了指导这些设计选择，作者提出了一个重要观点，“专用单元通过将数据存储结构与数据流和算法的数据局部性要求相调谐，实现了大部分的效率增益。”这一关键发现极大地有利于提高性能功耗比。卷积涉及大量的数据移动，例如，大图像必须从存储器访问。这些访问遵循明确的模式，只要确定卷积种类以及模板大小，就可以预先计算出这些模式。周全的设计将允许执行这些访问模式，以便最大限度降低多余访问，这不仅加快了计算速度，还大大降低了功耗。

总之，作者详细介绍了他们的研究，并展示了他们的设计在哪些方面填补了固定功能硬件和通用硬件之间的空白。具体而言，他们考虑了两种对立的方案，即自定义实现和通过 SIMD 指令增强的通用处理器。SIMD 指令以商业处理器中的相似功能为模型，如：Intel 的流式处理 SIMD 扩展和 ARM 的 NEON 扩展，它们的卷积引擎所实现的能源效率和面积效率是通过 SIMD 指令增强的通用处理器的 8-15 倍，但仅是定制化单元的 2-3 倍以内。该分析巧妙地证明了具有一定程度可编程性的专用化设计的利与弊。最后，单单这项分析就已值得研究，因为它清晰地解释了当今处理器中的功率和面积都消耗在什么地方。 C

Trevor Mudge (Email: tnm@eecs.umich.edu) 是密歇根大学安娜堡分校电气工程与计算机科学系 Bredt Family 工学教授。

译文责任编辑：张悠慧

DOI:10.1145/2735841

卷积引擎：平衡专用计算的效率与灵活性

作者：Wajahat Qadeer、Rehan Hameed、Ofer Shacham、Preethi Venkatesan、Christos Kozyrakis 以及 Mark Horowitz

摘要

通用处理器尽管可以用于多种用途，但这种灵活性的代价十分高昂，99% 以上的能源都浪费在可编程性开销上。我们发现，要想降低这一浪费，必须微调数据存储、计算结构以及二者与算法中的数据流和数据局部性模式之间的连接性。因此，通过回避全面可编程性，而以目标应用领域中的关键数据流模式为目标，我们就可以创建可在该域内的各类应用中编程并重复使用的高效引擎。

我们设计了一种卷积引擎 (CE)，它是一种专门适用于卷积类数据流的可编程处理器，这类数据流在计算摄影、计算机视觉及视频处理方面应用广泛。CE 的节能方式为捕获数据重用模式，消除数据传输开销，并支持在每次访问内存时执行大量运算。经证明，CE 的能源效率和面积效率是针对单内核优化的定制单元的 2-3 倍，是大多数图像处理应用使用的数据并行单指令多数据 (SIMD) 引擎的 8-15 倍。

1. 引言

无论是相对简单的嵌入式平台 RISC 核心，还是由数十亿晶体管组成的服务器 / 桌面计算机 CPU 芯片，处理器都是用途极其广泛的计算设备。处理器可以处理几乎所有类型的工作负载，包括 Web 应用程序、个人电子表格、图像处理工作负载、嵌入式控制应用、数据库以及财务应用等。此外，抽象编程和开发工具日益成熟，人类在编程领域积累数十年的宝贵知识，编码新应用已经变得非常容易。

然而，处理器也是一个效率低下的计算设备。通用处理器核心所消耗的大部分能源都来自预测、提取、解码、调度以及提交指令这些操作的开销。[2,7,16] 因此，与执行特定任务的专用硬件模块相比，处理器的能耗最多可高出 1000 倍。这些专用硬件模块的性能比处理器高几百倍，并且占用较小的硅面积。处理器尽管存在严重的低效弊端，但仍以多用途和可重用的优点而成为大多数计算系统的核心部件。

数十年来，得益于半导体设备的发展，我们提高了通用处理器的性能，并且避免其消耗过多能源。每一次新技术的出现都使逻辑门的切换能量呈指数级降低，我们因此能够创建规模更大、更复杂的设计，同时保证能耗量仅小幅增加。但近年来，由于能源缩放比例下降，[12] 我们无法再像以前那样提高处理性能。如今，要想在保证处理器能耗不变的情况下提高其性能，降低能源浪费是基本的解决办法。

本文将展示一种全新的高效处理架构，它支持计算摄影、图像处理和视频处理应用，我们将其命名为“卷积引擎 (CE)”。随着物美价廉的成像设备日益增多，计算摄影和计算机视觉应用有望在未来几年中成为重要的消费者计算工作负载。一些应用示例包括环境成像、手势控制、夜视和脉搏测量。

然而，这些应用的处理能力需求高达 TeraOps/s 级别，远远超出了通用处理器的能力，特别是能耗预算有限（小于 1 瓦特）的移动处理器。硬件加速器在计算效率方面拥有三个数量级的优势，因此当前移动系统使用由处理器和加速器组成的异构计算芯片。[11,15] 移动 SOC 中使用的视频编解码器硬件就是常见的加速器。然而，这些加速器或仅以单个算法为对象，或仅以稍加变化的算法变体为对象。今后的智能设备将拥有多种多样的应用集，需要以一种可编程的更好的计算解决方案来处理。我们的 CE 不但可以提供可编程性，还能达到与专用加速器接近的能耗。

我们发现，*专用单元实现最佳能效的方法是，微调数据存储结构与算法中的数据流和数据局限性模式*。我们的设计方法便以此为依据。这种微调可以消除冗余数据传输，有利于创建紧密耦合的数据通路和存储结构，从而针对每条指令和数据执行上百次低能耗运

本文的一稿标题为“卷积引擎：平衡专业计算的效率与灵活性”，刊登于《第 40 届计算机架构国际研讨会会议记录》（2013 年 6 月）。

算。如果处理器以目标应用领域内各类算法内核常用的计算图形和数据流模式为处理对象，就可以获得相似的节能收益。我们的 CE 使用通用 map-reduce 计算模式，可以描述图像处理领域的大量运算。这种设计所达到的能耗比通用处理器低两个数量级，是专用应用加速器的 2-3 倍。

下一节简要阐述了通用处理器能耗过高的原因以及现有优化策略的局限性。第 3 节介绍了卷积抽象和作为研究对象的五个应用内核。第 4 节描述了 CE 架构并重点讲解用于改善能源效率并 / 或支持灵活性和重用的功能。我们随后分别对比了 CE、带有 SIMD 扩展的通用核心以及分别适用于各个算法内核的若干个高度定制化解决方案在能源效率和面积效率方面的表现。第 5 节展示了 CE 在处理大多数应用时的性能是定制单元的 2-3 倍，是 SIMD 解决方案的 10 倍。

2. 背景

图 1 解释了通用处理器能效低下的原因，并比较了各类算术运算的能耗与极其简单的 RISC 处理器的整体指令能耗。执行有用计算工作的算术运算的能耗仍然低于指令提取、解码、线路管理、程序定序等指令开销所浪费的能源。媒体处理应用的开销更为严重，此类应用往往处理短数据，每次运算仅需要 0.2–0.5 pJ (90 nm) 的能源，其结果是 99% 的能源都用于其它开销。

必须大幅降低开销，才能提高能源效率。处理器设计因而面临两项制约因素，(i) 处理器在处理每个指令时需要执行上百次运算才足以摊销指令开销，(ii) 处理器还必须提取小数据，即使每次都是缓存命中也会消耗 25 pJ (90 nm) 的能量，相比之下，算术运算的开销只有 0.2–0.5 pJ。

这两项制约因素看似相互矛盾，因为在每个周期执行上百次运算时，往往都需要从内存中读取大量数据。如果运算中的大多数指令可以处理由先前指令产生的中间结果或重用先前指令所用的大部分输入数据，那么情况就会有所缓解。如果拥有充足的存储结构将“历史数据”保留在处理器数据通路中，那么处理器在处理每个指令时都可以执行上百次的运算，并且无需频繁地访问内存。幸运的是，计算限制应用中的图像处理和视频处理算法正好可以解决这两项制约因素，因为二者可支持数据并行和数据重用。

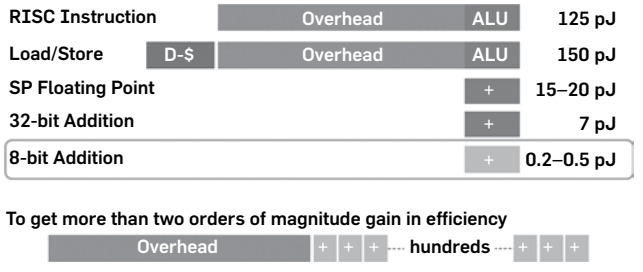
目前，SIMD 单元已内置到大部分高性能处理器，这是针对计算密集型应用的最有效的通用优化手段。SIMD 单元通常可以将能耗降低一个数量级，方法为在单个循环中同时处理很多数据运算对象（通常为 8-16）。但是，正如 Hameed 等人[7] 所论述的那样，内置 SIMD 单元所带来的能效仍然较专用硬件加速器低两个数量级，这是因为 SIMD 模型无法扩展并适应更高程度的数据并行。

为了更好地了解传统 SIMD 单元的架构局限性，请考虑列表 1 所示的针对 16 位 8x8 区块的二维绝对误差和运算 (SDA)。[5] 2D SAD 运算在 H.264 视频解码器等多媒体应用中使用广泛，用于在参考图或视频帧中查找 2D 图像子区块的最近接匹配。列表 1 针对参考帧的 *srchWinHeight* × *srchWinWidth* 搜索窗口内的每个位置执行了搜索，最终得到 4 个嵌套循环。这 4 个嵌套循环是相互独立的，可以同时并行化。此外，每个 SAD 结果都可以充分重用用于计算先前的输出的输入数据，无论是垂直方向还是水平方向。

但是，常见的 SIMD 单元的寄存器大小为 128 位，只能处理适合单个寄存器的元素，将并行性限制在最内部的循环。尝试提高 SIMD 宽度以增加并行性，必须同时从寄存器堆读取多个图像行（在第二靠近内部的嵌套上实现并行），或同时读取图像数据的多个重叠行（在多个水平输出上实现并行）。但 SIMD 模型不支持这两种操作。

要想借助数据重用，则必须将完整的 8x8 区块存储到 SIMD 寄存器堆的 8 个行中，以便在本地使用这些数据来计算后续的输出像素。这在垂直方向很容易

图 1. 对比功能单元与 90nm 常见 RISC 指令的能耗。摊销处理器开销的策略，包括在处理每个指令时执行上百个低能耗运算。



列表 1. 2D 8 × 8 绝对误差和运算，常见用于 H.264 运动估计。

```
for (sWinY = 0; sWinY < srchWinHeight; sWinY++){
  for (sWinX = 0; sWinX < srchWinWidth; sWinX++){
    sad = 0;
    for (y = 0; y < 8; y++){
      for (x = 0; x < 8; x++){
        cY = y + sWinY; cX = x + sWinX;
        sad += abs(ref[cY][cX] - cur[y][x]);
      }
    }
    outSad[sWinY][sWinX] = sad;
  }
}
```

做到，因为每个新输出只需要一个新行，七个旧行可以重用。但是，要想在水平方向做到重用，八个行必须先分别移动一个像素，然后才能计算每个新的输出像素，这会产生巨大的指令开销。由于移动过程会破坏旧像素，所以重用只能在垂直方向或水平方向实现，而不能在两个方向同时实现，这导致每个数据项都会从内存中提取八次，浪费了过多的内存能量。

GPU 可实现更高程度的并行化，因为它使用大量的简单 SIMD 核心，每个核心都带有本地寄存器资源和庞大的内存带宽，可以保持较高的计算吞吐量。在计算吞吐量提高的同时，能源消耗也显著增加，这是由于数据访问开销过大所致。我们曾使用 GPUGPUSim 模拟器 [1] 以 GPUWattch 能源模型 [9] 来衡量基于 H.264 SAD 算法 [13] 的优化 GPU 实现的性能和能耗，从而断定该 GPU 实现的运行速度和能耗分别是嵌入式 128 位 SIMD 单元的 40 倍和 30 倍。这进一步表明了，最大限度较少内存访问，并实现低开销的本地数据访问势在必行。

正如下一节所论述的那样，SAD 运算属于一大类算法，与卷计算法类似，拥有理想的并行和重用特征，执行效率较高。下一节将探讨该计算抽象并列举一些应用实例。

3. 计算模型

方程式 (1) 和方程式 (2) 定义了标准离散 1D 和 2D 卷积。在处理图像时，*Img* 是从图像位置导出像素值的函数，而 *f* 是应用于图像的滤波器。实用内核通过缩小过滤器的大小来减少计算量，2D 卷积的顺序通常是从 3x3 到 8x8，但这种方法会略微影响计算准确性：

$$(Img * f)[n] \stackrel{\text{def}}{=} \sum_{k=-\infty}^{\infty} Img[k] \cdot f[n-k] \qquad 7.$$

$$(Img * f)[n,m] \stackrel{\text{def}}{=} \sum_{l=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} Img[k,l] \cdot f[n-k,m-l] \qquad 7.$$

我们通过确定卷积的两个组成部分来概括卷积的概念，这两个部分分别是：map（映射）运算和 reduce(规约) 运算。在方程式 (1) 和方程式 (2) 中，映射运算是对各对像素和抽头系数进行的乘法，而规约运算是对到 $[n, m]$ 位置的输出值的所有像素和抽头系数对的求和。将方程式 (2) 中的映射运算的 $x \cdot y$ 替换为 $|x-y|$，但仍将规约运算保留为求和，这样就可以得到用于 H.264 运动估计的绝对误差和 (SAD) 函数。将规约运算中的 Σ 替换为 *max* 就可以得到最大绝对差的运算方法。方程式 (3) 表达了由我们所设计的 CE 的计算模型，其中 *f*、*Map* 和 *Reduce*（即其中的 *R*）都是伪指令，而 *c* 代表卷积的大小：

$$(Img \overset{CE}{*} f)[n,m] \stackrel{\text{def}}{=} R_{\|l\|<c}\{R_{\|k\|<c}\{Map(Img[k], f[n-k,m-l])\}\} \qquad 7.$$

很多应用都使用卷积类数据流，但其局限性在于规约运算中仅存在单个满足结合律的运算。一些应用采用与卷积类似的数据局部性模式，但需要通过专门的图运算而不是简单的规约运算来完成。CE 可以用于这些应用，需要提高*规约*运算的复杂性以支持非交换函数，从而在每个规约阶段使用不同函数。这一泛化的组合网络扩展了“规约”阶段，以创建可输入大量值并以高效融合的超级指令计算少量输出的结构（参见图 2）。

扩展存在一定弊端，将输入置入组合树会带来显著影响，因此，要充分发挥新的泛化的*规约*运算的能力，数据供应网络必须具备高度灵活性，可将所需数据移动到正确的位置。为此，可以扩展*映射*运算的定义，以支持数据置换网络，同时继续支持已受支持的计算运算集。这些改进提高了*映射*和*规约*运算的可推广性和可用性，从而扩大了二者的应用空间。

接下来，我们将介绍研究中所使用的几个内核，并探讨它们如何映射到上文所定义的计算抽象。图表 1 总结了这些信息。

3.1. 运动估计

运动估计是 H.264 等众多视频编解码器的关键组成部分，消耗了将近 90% 的 H.264 软件实现执行时间。[4, 7] 内核负责处理视频帧的子区块，尝试找到每个子区块在视频流中的先前和 / 或后续参考帧上的位置。请注意，H.264 的运动估计分两步计算：

整数像素运动估计 (IME)： IME 使用第 2 节中提及的 SAD 运算来查找参考图像中的匹配区块。请注意，SAD 本身就适用于 CE 抽象：*映射*函数计算绝对差，*规约*函数求和。

图 2. 泛化的规约单元将多个运算融合到一个超级指令中。

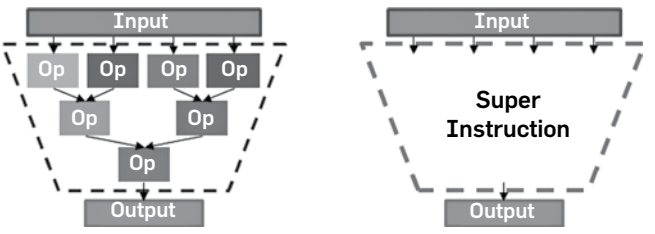


表 1. 将内核映射到卷积抽象。

	映射	规约	模板大小	数据流
IME SAD	绝对差	加	4 × 4	2D 卷积
FME 1/2 像素升采样	乘	加	6	1D 水平和垂直卷积
FME 1/4 像素升采样	平均	无	–	2D 矩阵运算
SIFT 高斯模糊	乘	加	9, 13, 15	1D 水平和垂直卷积
SIFT DoG	减	无	–	2D 矩阵运算
SIFT 极值	比较	逻辑与	9 × 3	2D 卷积
Demosaic 插值	乘	复杂	3	1D 水平和垂直卷积

减法等内核运算单个像素，因此没有定义模板大小。我们称之为矩阵运算。此类运算不存在规约步骤。

分数像素运动估计 (FME)： FME 将 IME 步骤获得的初始匹配改善至 1/4 像素分辨率。FME 先对 IME 所选区块进行升采样，然后执行稍加修改的 SAD 运算变体。升采样同样适用于卷积抽象，并且由两项卷积运算组成：(1) 使用 6 抽头 2D 可分滤波器对图像块进行升采样。这部分是纯卷积运算。(2) 以 2 为因子对相邻像素插值，对所获得的图像进行上采样，该运算可以定义为不带*规约*的*映射*运算（用于生成新像素）。

3.2.SIFT

尺度不变特征转换 (SIFT) 用来查找图像中的区别性特征。[10] 为确保图像尺度不变，需要对图像进行高斯模糊和下采样，以便创建尺度越来越粗的图像金字塔。然后计算每两个相邻图像尺度之间的差别，就可以构建高斯差 (DoG) 金字塔。通过在 DoG 金字塔上查找尺度空间极值，就可找出特征值。[10]

高斯模糊和降采样从本质上来说属于 2D 卷积运算。查找尺度空间极值需要进行 3D 模版 (stencil) 计算，但我们可以将该运算转化为 2D 模版计算，方法为将不同图像的行交叉存取到单个缓冲区中。以比较作为*映射*运算，以逻辑与作为*规约*运算，将极值运算映射到卷积运算。

3.3.Demosaic

相机传感器往往输出以贝尔模式布局的红色、绿色和蓝色 (RGB) 马赛克图案。[3] 在每个位置，缺少的两种颜色都采用周围网格的亮度值和色彩值进行插值。由于色彩信息抽样不足，插值的难度较大，任何线性方法都会造成色晕。我们使用的是基于自适应颜色层插值 (ACPI) 的 Demosaic 实现，[8] 先计算图像梯度然后在最小梯度方向使用 3 抽头滤波器。尽管该方法适用于泛化的卷积流，但需要使用复杂的“规约”树来实现基于梯度的选择。数据访问模式也是较重要的，因为在执行插值前，马赛克中的每个色彩值都必须分离开来。

4. 卷积引擎

卷积运算是高度计算密集型的运算，在处理大型模板时更是如此。卷积运算支持数据并行，所以适用于向量处理。但是，如上文所述，现有 SIMD 单元由于寄存器堆的组织方式而无法充分利用卷积所固有的并行性和局部性。CE 借助移位寄存器结构，打破了这些限制。图 3 列举了一个 2D 卷积的应用实例，如果存储结构能够针对输入数据生成多个移位版本，那么就能从小型的 16 × 8 2D 寄存器填充 128 个算逻运算部件，并消耗较低的访问能源和芯片面积。1D 水平卷积和 1D 垂直卷积也能获得相似的收益。接下来，我们会发现 CE 可以通过创建第 3 节中提及的融合超级指令来进一步降低开销。

CE 是被作为 Tensilica 可扩展 RISC 处理器内核的专用扩展硬件开发的。[6] 该扩展硬件使用 Tensilica’s TIE 语言开发。[14] 下一节将探讨 CE 扩展硬件的关键模块，请参见图 4。

4.1. 寄存器堆

2D 移位寄存器可用于垂直和 2D 卷积流，并且支持垂直行移位，即当 2D 模板垂直向下移动到图像中时，一个新的像素数据行就会移位进来。2D 移位寄存器可以同时访问它的所有元素，支持接口单元向 ALU 馈送任何数据元素。1D 移位寄存器用来向水平卷积流提供数据。当 1D 模板上移一个图像行后，新图像像素会水平移位到 1D 寄存器中。

2D 系数寄存器存储不会随着模板在图像移动而改变的数据。此类数据包括滤波器系数、用于执行 SAD 的 IME 中的当前图像像素、窗口化最大 / 最小模板中央的像素。卷积运算的结果会重新写入 2D 移位寄存

图 3. 实现 8x8 2D SAD 运算，利用列表 1 中的四个循环的并行性。参考块位于 2D 移位寄存器，而当前块存储在 2D 寄存器。由于这两种寄存器均支持 8x8 块的 2D 访问，所以 64 ALU 可以并行运算。为启用更高程度的并行性，并利用水平方向的数据重用，移位寄存器生成了数对多个重叠 8x8 块，它们随后通过多路复用器馈送到 ALU。这些对支持并行执行 128 ALU，并且并行生成 2 个输入。在生成 4 对水平输入后，移位寄存器会上移一行，为新一行搜索窗口提供空间，以实现垂直数据重用。

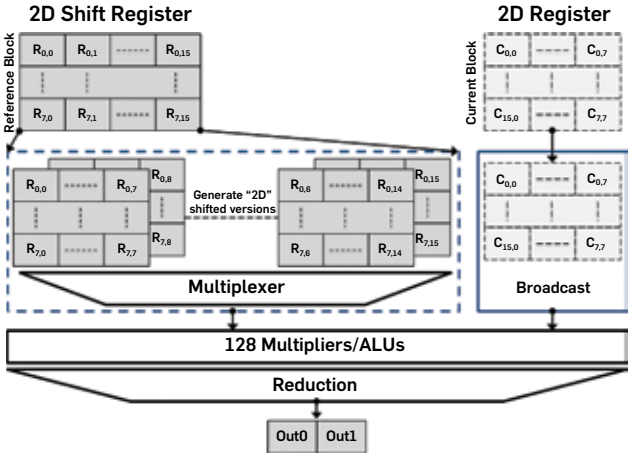
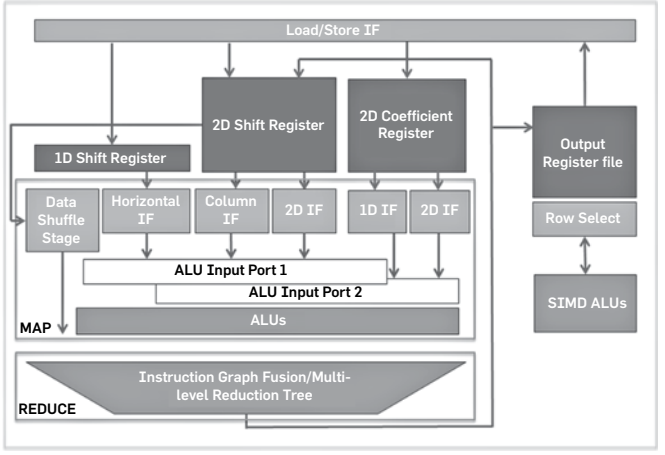


图 4. 卷积引擎块图解。接口单元 (IF) 将寄存器堆连接到功能单元，并提供移位广播来促进卷积。结合了指令图形融合 (IGF) 阶段的数据洗牌 (DS) 阶段会创建泛化的规约单元，并且被称作复杂图形融合单元。



器或输出寄存器。后续章节将展示数据寄存器堆如何用作向量寄存器堆来处理图 4 中的向量单位。

4.2. 映射和规约逻辑

如前所述，我们将卷积抽象为可将输入像素转化为输出像素的*映射*和*规约*步骤。在我们的实现中，接口单元和 ALU 协同工作以实现*映射*运算，接口单元会对特定映射模式的所需数据以及执行相应数学运算的功能单元进行排列。

接口单元。接口单元 (IF) 将来自寄存器堆的数据排列成执行映射运算所需的特定模式。对于 2D 卷积，可从 2D 寄存器同时访问多个移位 2D 子块。支持的块大小包括 2 × 2、4 × 4、8 × 8 等，并且以卷积内核大小作为选择依据。同样，对于垂直卷积，可以并行访问多个 2D 寄存器列，并且支持多种列访问大小。最后，对于水平卷积，1D IF 支持从 1D 移位寄存器访问多个移位的块。我们还尝试利用更泛化的排列层来支持任意映射。

功能单元。由于接口单元负责处理所有数据重排列，所以功能单元只是一列简短的二输入定点运算 ALU。我们还支持使用乘数和绝对差，来简化 SAD 和其他类型算术运算，如加法、加法和比较。ALU 的结果会馈送到规约阶段。

规约单元。*映射 - 规约*运算中的*规约*部分由可编程规约阶段处理。根据应用的需求，目前支持算术和逻辑规约阶段。规约程度将取决于内核大小，比如 4x4 2D 内核需要按照 16:1 进行规约，而 8 抽头 1D 内核需要按照 8:1 进行规约。因此，规约阶段作为组合树来实现，结果从组合树的多个阶段得出。

为了能够创建第 3 节所描述的“超级指令”，我们对组合树进行了扩展，即在组合树不同层级加入各类算术运算支持，使之可以处理非交换运算。该融合减少了所需指令的数量，消除了寄存器堆中的中间数据临时存储，从而提高了计算效率。由于这一复杂的数据组合不必是具备交换性的，因此，每次输入到组合网络的数据（映射运算的结果）都必须正确无误。因此，CE 还加入了“数据洗牌阶段”，即利用灵活的重排网络对输入数据进行排列。

4.3. 其他硬件

为便于对卷积结果进行向量运算，我们还添加了一个 32 元素 SIMD 单元。该单元与 2D 输出寄存器相接，并将后者用作向量寄存器堆。该单元较常用 SIMD 单元更宽泛，它可以处理由卷积数据通路生成的中间数据，因此不会受限于数据内存访问。尽管如此，向量单元仍属轻量级，因为它仅支持基本的向量加减类型的运算，并且无法支持乘法等高开销运算。

应用可能会执行既不符合卷积块也不符合向量单元的计算，或者以其他方式受益于固定函数实现。如果设计人员希望构建适用于此类计算的定制化单元，CE 可以支持固定函数块来访问输出寄存器堆。该模型类似于 GPU。在 GPU 中，自定义块用于实现光栅化，

并与渲染核心共同工作。对于这些应用，我们创建了三个自定义函数块，用于计算 IME、FME 以及 FME 中 Hadamard 转换的运动矢量开销。

4.4. 资源大小调整

目标应用的能效因素和资源要求推动了 CE 内部各类资源的增加。正如 Hameed 等人所论述的那样，[7] 要想摊销指令开销，媒体处理应用必须在处理每个指令时执行上百次的基于短 8 位加 / 减运算的 ALU 运算。但是，多数卷积流应用都基于能耗更高的乘法运算。我们的分析显示，对于基于乘法的算法，在处理每个指令时执行 50-100 个运算就足以充分摊销指令开销。如果 ALU 的数量继续大幅提高，收益递减现象就会出现，使这些单元保持繁忙所需的存储规模就会增加，因此，存储空间和数据访问能耗将升高。在研究中，我们选取的 ALU 阵列大小为 28，并相应地重置了资源，以确保这些 ALU 保持繁忙。为进一步提高灵活性，我们允许关闭一半 ALU 阵列和计算结构。每个资源的大小和能力显示在图表 2 中。这些资源支持的滤波器大小为 4×4、8×8（1D 滤波）以及 16×16（2D 滤波）。请注意，寄存器堆大小以 2 的幂数偏离，以高效处理卷积运算中常见的边界条件。

4.5. 对卷积引擎编程

CE 作为处理器的扩展来实现，并且向处理器 ISA 添加了一小组指令。这些 CE 指令可在必要时通过编译器内部以常规的 C 代码形式被使用。图表 3 列举了主要的 CE 指令。配置指令用于设置内核参数，如卷积大小、映射和规约阶段使用的 ALU 运算等。然后是每个寄存器资源都会进行的加载和存储运算。最后是计算指令，它们分别应用于三种受支持的卷积流（1D 水平、1D 垂直和 2D）。以 CONVOLV_2D 指令为例，它会从 2D 和系数寄存器中读取一组值，然后执行卷积运算，并将结果写入 2D 输入寄存器的 0 行。

表 2.CE 中各类资源的大小。

	资源大小
ALU	128 × 10 位 ALU
1D 移位寄存器	80 × 10 位
2D 输入移位寄存器	16 行 × 36 列 × 10 位
2D 输出移位寄存器	16 行 × 36 列 × 10 位
2D 系数寄存器	16 行 × 16 列 × 10 位
水平接口	4、8、16 核模式
垂直接口	4、8、16 核模式
2D 接口	4×4、8×8 和 16×16 模式
规约树	4:1, 8:1, …, 128:1

列表 2 中列举的代码包含了主要的 CE 指令，并实现了 2D 8x8 滤波器。首先，CE 被设置为在映射阶段执行乘法，在规约阶段执行加法，这是滤波要求的设置。然后，对卷积大小进行设置，这决定了从寄存器向 ALU 馈送数据的模式。接下来，将抽头系数载入到系数寄存器中。最后，主处理循环不断将新的输入像素载入到 2D 移位寄存器，并发出 2D_CONVOLVE 运算指令来执行滤波。尽管每次载入会读取 16 个新像素，但我们的 128-ALU CE 被配置为每次运算只能

表 3. 添加到处理器 ISA 的主要指令。

	描述
SET_CE_OPS	设置映射和规约步骤的算术函数
SET_CE_OPSIZE	设置卷积大小
LD_COEFF_REG_n	将 *n* 位载入到 2D 系数寄存器的特定行
LD_1D_REG_n	将 *n* 位载入到 1D 移位寄存器。可选择向左移位
LD_2D_REG_n	将 *n* 位载入到 2D 移位寄存器。可选择向下移位
ST_OUT_REG_n	将 2D 输入寄存器的首行存储到 2D 输出寄存器
CONVOLVE_1D_HOR	1D 卷积步骤 - 从 1D 移位寄存器输入
CONVOLVE_1D_VER	1D 卷积步骤 - 列访问 2D 移位寄存器
CONVOLVE_2D	2D 卷积步骤，2D 访问 2D 移位寄存器

列表 2.C 代码示例，针对垂直图像条纹实现 8x8 2D 滤波器并向每个输出加 2。

```
// Set MAP function = MULT, Reduce function = ADD
SET_CE_OPS (CE_MULT, CE_ADD);

// Set convolution size 8
SET_CE_OPSIZE(8);

// Load eight rows of eight 8-bit coefficients
// into Coeff Reg's rows 0 to 7
for(i = 0; i < 8; i++){
    LD_COEFF_REG_128(coeffPtr, 0);
    coeffPtr += coeffWidth;
}

// Load & shift seven rows of sixteen input pixels
// into 2D shift register
for(i = 0; i < 7; i++){
    LD_2D_REG_128(inPtr, SHIFT_ENABLED);
    inPtr += width;
}

// Filtering loop
for (y = 0; y < height; y++) {
   // Load & Shift 16 more pixels
   LD_2D_REG_128(inPtr, SHIFT_ENABLED);

   // Filter first 8 locations. Because we have
   // access to 128-ALUS, we can filter two 8x8
   // blocks in parallel
   for(RW_OFFSET = 0; RW_OFFSET < 8; RW_OFFSET+=2){
      CONVOLVE_2D(RW_OFFSET, RW_OFFSET);
   }

   // Add 2 to row 0 of output register
   SIMD_ADD_CONST (0, 2);

   // Store 8 output pixels
   ST_OUT_REG_64(outPtr);

   inPtr += width;
   outPtr += width;
}
```

处理 2 个 8x8 滤波运算。因此，每次迭代会执行 4 个 2D_CONVOLVE 运算。为便于说明，我们加入了一个 SIMD 指令，其内容为向卷积运算得到的每个输出值加 2。输出寄存器的结果重新写入内存。

图 5 将该执行过程映射到 CE 硬件上。8x8 系数被存储在系数寄存器堆的前 8 个行。8 个行的图像数据移位至 2D 移位寄存器的前 8 个行。由于我们使用 128 功能单元，我们可以同时滤波 2 个 8x8 2D 位置。因此，2D 接口单元生成两个移位版本的 8x8 块，后者被重新列入 1D 数据并馈送给 ALU。功能单元使用相应的系数对每个输入像素执行元素级乘法，所得结果会馈送到规约阶段。规约程度取决于滤波器大小。在本例中，滤波器大小为 8x8，因此选择按 64:1 进行规约。规约阶段的两个输出经归一化处理后，写入输出寄存器。

值得注意的是，与独立加速器不同，CE 的运算顺序完全由 C 代码控制，使算法具备全面的灵活性。比如，在上述滤波代码中，该算法可以向内存生成 1 个 CE 输出，然后对该输出执行一系列非 CE 运算，此后再调用 CE 生成其他输出。

5. 评估

为了评估 CE 的效率，我们将第 3 节所描述的三个目标应用映射到由 2 个 CE 组成的多核处理器 (CMP)。为了量化该可编程单元的性能和能耗，我们还为这三个应用分别构建了定制化异构多核处理器 (CMP)。这些定制化 CMP 基于专用的核心，是适应于相应用途的高度专业化内核。CE 和专用核心都使用 Tensilica TIE 语言构建，用作处理器核心的数据通路扩展。[14]Tensilica TIE 编辑器根据该描述为各扩展处理器配置生成了模拟模型和 RTL。

为快速模拟并评估 CMP 配置，我们创建了多处理器模拟框架。该框架使用 Tensilica Xtensa 建模平台 (XTMP) 对处理器和缓存执行周期精确的模拟。我们使用 Tensilica 能源资源管理器工具来估算能耗，该工具利用程序执行跟踪来详细分析处理器核心和内存系统的能耗。估算的能耗在实际能耗的 30% 以内。我们为 CMP 制作了平面图，并估算了接线能耗，以便将互连能耗考虑在内。随后，我们将互连能耗与 Tensilica 工具测出的能耗估值相加。模拟结果使用 90nm 技术，1.1V 工作电压，目标频率是 450 MHz。所有单元都正确地连接在一起，以达到目标频率。

图 6 和图 7 比较了 CE、128 位数据并行 (SIMD) 引擎、定制化加速器实现在执行五种算法上的性能和能耗。大多数情况下，我们将 SIMD 引擎用作 16 通道、8 位数据通路，但在一些例子中，我们创建了 8 通道、

图 5. 在 CE 上执行 8x8 2D 滤波器灰色方框代表单元未在例子使用。

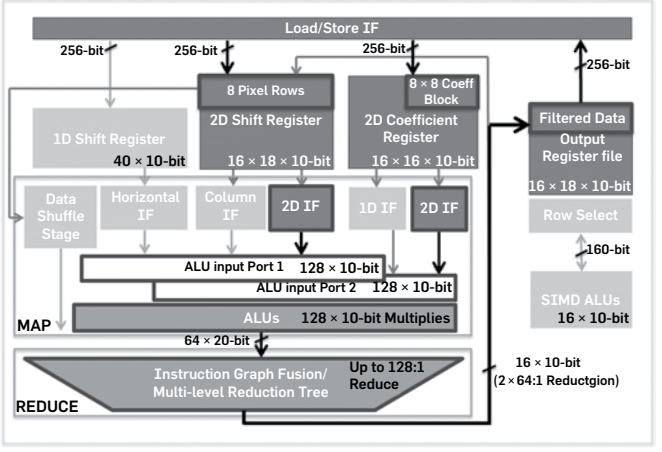


图 6. 能耗归一化到定制化实现。

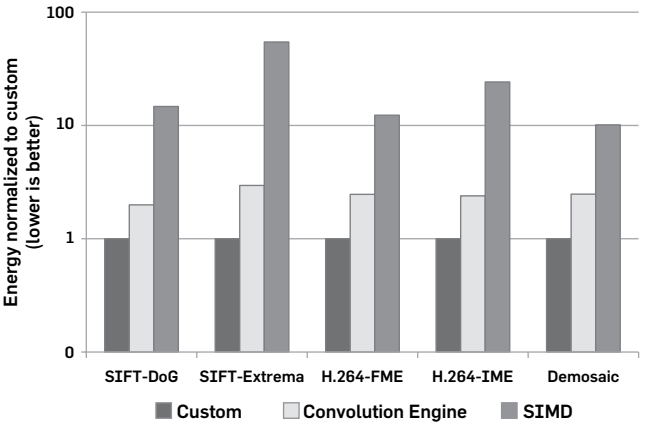
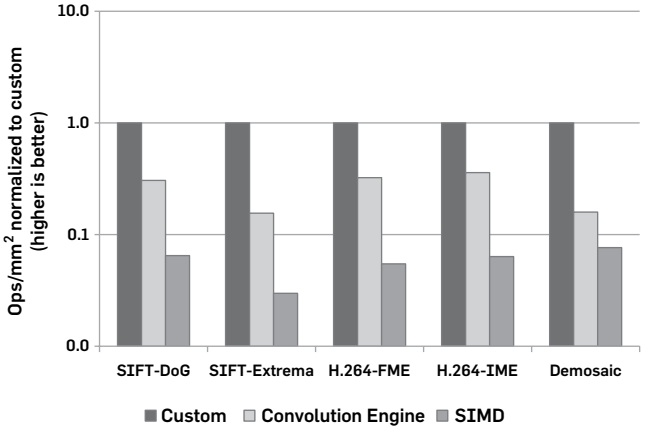


图 7.Ops/mm² 归一化到定制化实现：每个核心在每秒钟处理的图像块的数量，除以核心的面积。对于 H.264，图像块是 16×16 宏块；对于 SIFT 和 Demosaic，图像块是 64×64 图像块。



16 位数据通路。就算法而言，提高该单元的宽度不会显著改变能效。

固定函数数据点确实凸显了定制化的优势：对于每种应用，定制化加速器的能耗都比 SIMD 引擎低 5-50 倍。定制化加速器在每单位面积上的性能比 SIMD 实现高 8-30 倍。Demosaic 的改进最不明显（8 倍），因为每当它从内存加载一个像素值，就会生成两个新像素值。因此，发生在定制化计算运算后的加载 / 存储及地址操作运算都会成为瓶颈，在全部指令中所占比例将近 70%。

请注意，IME 和 SIFT 极值计算中的收益最大。两种内核都依赖于能耗较低的短整数*加 / 减*运算（相对于滤波和升采样中的*乘法*）。我们在上文讨论过，SIMD 实现的指令开销和数据访问能耗与相应的计算量相比仍然非常庞大。而定制化加速器可在相应的内核中利用并行性和数据重用，从而充分摊销指令和数据提取开销。

现在，我们可以更好地了解 CE 的地位。CE 的架构与基于卷积算法的数据流高度匹配，因此固定功能单元与 CE 在指令流方面的差异非常小。与 SIMD 实现相比，CE 的能耗低 8-15 倍，但 Demosaic 应用除外，其中的改进程度为 4 倍，CE 的性能与面积比率提高了 5-6 倍。Demosaic 的收益仍然最低，这是由于载入和存储过量所致。如果我们不考虑内存运算对 Demosaic 的影响，假设其输出被输入到图像管道中的另一与类似于卷积的阶段，那么基于 CE 的 Demosaic 的优势是 SIMD 的近 7 倍，是定制化加速器的 6 倍以内。能源比高于定制化实现，反映了一般性规约中更灵活通信的成本。

但是，对于其他应用，CE 的能量开销是固定功能单元的 2-3 倍，而面积开销仅为 2 倍。尽管这些开销微不足道，但为了解效率低下的原因，图 8 和图 9 针对每种应用创建了三个不同实现：定制化实现、带有灵活寄存器的固定计算、完全灵活的 CE。

这两张图显示了，当通信路径变得复杂时，能效受到的影响最为严重。如果基本计算的能耗较低，开销就会变得更加严重。一般而言，通信路径的复杂性会随着存储结构的大小而增加，因此可编程单元中必不可少的预留空间寄存器会降低能效。由于此类结构在路由和复用时对逻辑性要求较高，并且对能量释放有直接影响，因此能量开销进一步增加。更灵活的功能单元的开销较小，能以低开销提供灵活性。

图 8. 随着核心的可编程性递增，能耗会发生变化。

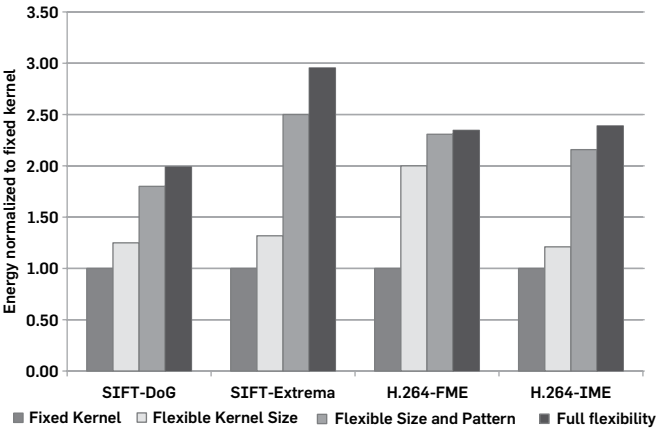
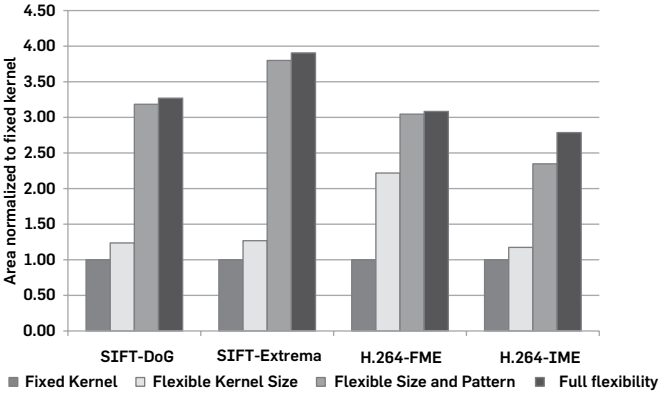


图 9. 随着核心的可编程性递增，面积会相应地增加。



6. 结论

随着专业分工成为解决当前架构能效制约的主要途径，充分利用这些专用引擎已迫在眉睫。与此同时，专用引擎的灵活性和易用性也不容忽视。尽管灵活的专用引擎听起来很矛盾，但我们发现，以范围较大的目标应用领域中关键数据流和数据局部性模式为重点，可以构建能效较高、用户可编程的引擎。我们所展示的 CE 支持多种基于卷积类模式的算法，可用于计算摄影、图像处理和视频处理。只需一个 CE 设计就能支持基于各种尺寸、维度和计算类型的卷积的应用。CE 的节能方式为捕获数据重用模式，消除数据传输开销，并支持在每个循环时执行大量运算。CE 的能源效率和面积效率是单内核加速器的 2-3 倍，是大多数应用使用的带 SIMD 扩展通用核心的 8-15 倍。尽管 CE 只是一个例子，但我们希望其他应用领域也会开展类似研究，从而推出其他高效、可编程的专用加速器。

鸣谢

本文借鉴了美国国防部高级研究计划局的研究成果，协议编号 HR0011-11-C-0007。本文所阐述的见解、结论或建议均属作者个人行为，不能代表美国国防部高级研究计划局的观点。

参考资料

1. Bakhoda, A., Yuan, G., Fung, W.W.L., Wong, H., Aamodt, T.M.Analyzing CUDA workloads using a detailed GPU simulator.In *ISPASS:IEEE International Symposium on Performance Analysis of Systems andSoftware* (2009).
2. Balfour, J., Dally, W., Black-Schaffer, D., Parikh, V., Park, J. An energy-efficient processor architecture for embedded systems. *Comput.Architect.Lett.7*, 1 (2007), 29–32.
3. Bayer, B. *Color Imaging Array*.US Patent Application No.3971065 (1976).
4. Chen, T.-C., Chien, S.-Y., Huang, Y.-W., Tsai, C.-H., Chen, C.-Y., Chen, T.-W., Chen, L.-G. Analysis and architecture design of an HDTV720p 30 frames/sec H.264/AVC encoder. *IEEE Trans.Circuits Syst.Video Technol.16*, 6 (2006), 673–688.
5. Corbal, J., Valero, M., Espasa, R. Exploiting a new level of DLP in multimedia applications.In *Proceedings of the 32nd Annual International Symposium on Microarchitecture* (Nov. 1999), 72–79.
6. Gonzalez, R. Xtensa:A configurable and extensible processor.*Micro IEEE 20*, 2 (Mar. 2000), 60–70.
7. Hameed, R., Qadeer, W., Wachs, M., Azizi, O., Solomatnikov, A., Lee, B.C., Richardson, S., Kozyrakis, C., Horowitz, M. Understanding sources of inefficiency in general-purpose chips.In *ISCA '10:Proceedings of the 37th Annual International Symposium on Computer Architecture* (2010), ACM.
8. Hamilton, J.F., Adams, J.E.*Adaptive Color Plane Interpolation in Single Sensor Color Electronic Camera*.US Patent Application No.5629734 (1997).
9. Leng, J., Gilani, S., Hetherington, T., Tantawy, A.E., Kim, N.S., Aamodt, T.M., Reddi, V.J.GPUWattch:Enabling energy optimizations in GPGPUs.In *ISCA 2013:International Symposium on Computer Architecture* (2013).
10. Lowe, D. Distinctive image features from scale-invariant keypoints.*Int.J.Comput. Vision 60*, 2 (2004), 91–110.
11. NVIDIA Inc.Tegra mobile processors. http://www.nvidia.com/object/tegra-4-processor.html.
12. Shacham, O., Azizi, O., Wachs, M., Qadeer, W., Asgar, Z., Kelley, K., Stevenson, J., Solomatnikov A., Firoozshahian, A., Lee, B., Richardson, S., Horowitz, M. Rethinking digital design:Why design must change.*IEEE Micro 30*, 6 (Nov. 2010), 9–24.
13. Stratton, J.A., Rodrigues, C., Sung, I.-J., Obeid, N., Chang, L.W., Anssari, N., Liu, G.D., Hwu, W.-M.W.Parboil:A Revised Benchmark Suite for Scientific and Commercial Throughput Computing.IMPACT Technical Report.In *IMPACT-12-01*, 2012.
14. Tensilica Inc.Tensilica Instruction Extension (TIE) Language Reference Manual.
15. Texas Instruments Inc.OMAP 5 platform. www.ti.com/omap.
16. Venkatesh, G., Sampson, J., Goulding, N., Garcia, S., Bryksin, V., Lugo-Martinez, J., Swanson, S., Taylor, M.B.Conservation cores:Reducing the energy of mature computations. In *ASPLOS'10* (2010), ACM.

Wajahat Qadeer 和 Rehan Hameed ([wqadeer, rhameed]@gmail.com)，加利福尼亚州帕洛阿尔托。

Ofer Shacham (shacham@alumni.stanford.edu)，Google，加利福尼亚州山景城。

Preethi Venkatesan (preethiv@stanford.edu)，Intel 集团，加利福尼亚圣克拉拉。

Christos Kozyrakis 和 Mark Horowitz ([kozyraki, horowitz]@stanford.edu)，斯坦福大学，加利福尼亚。

译文责任编辑：张悠慧