



@ken_power



Second International Workshop on Software
Architecture and Metrics



A Metric-Based Approach to Managing Architecture-Related Impediments in Product Development Flow An Industry Case Study from Cisco

Ken Power
Principal Engineer
Service Provider Video Software and Solutions
Cisco Systems

May 16th 2015

A Metric-Based Approach to Managing Architecture-Related Impediments in Product Development Flow
An Industry Case Study from Cisco

Ken Power
Cisco Systems, Inc.
Galway, Ireland
ken.power@gmail.com

Kieran Conboy
National University of Ireland, Galway
Galway, Ireland
kieran.conboy@nuigalway.ie



im·ped·i·ment  *noun* \im-'pe-də-mənt\

: something that makes it difficult to do or complete

something : something that interferes with movement or progress



We want early warning **leading indicators** to identify architecture-related problems.

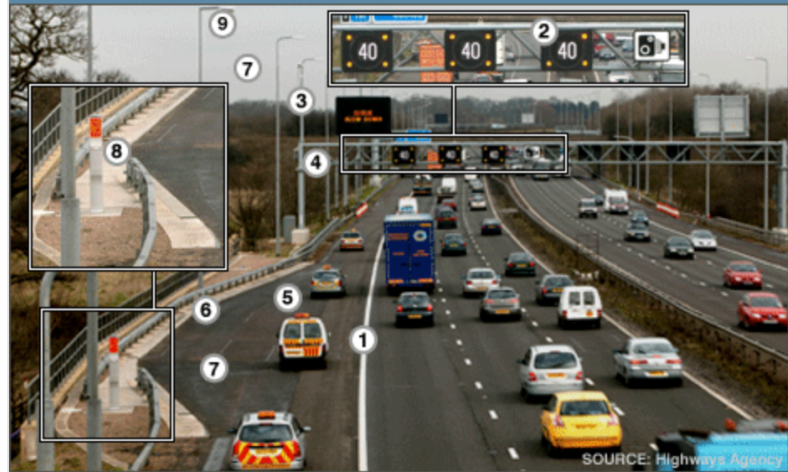
By focusing on two questions, we can get a good understanding of where impediments influence architecture, and vice versa:

1. **How does work (value) flow** through your team and organization?
2. What is **impeding the flow** of work (value)?

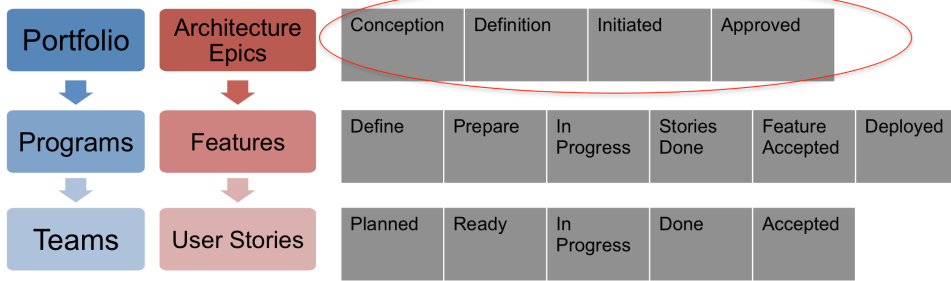
Architecture can be an enabler of flow



ACTIVE TRAFFIC MANAGEMENT SYSTEM

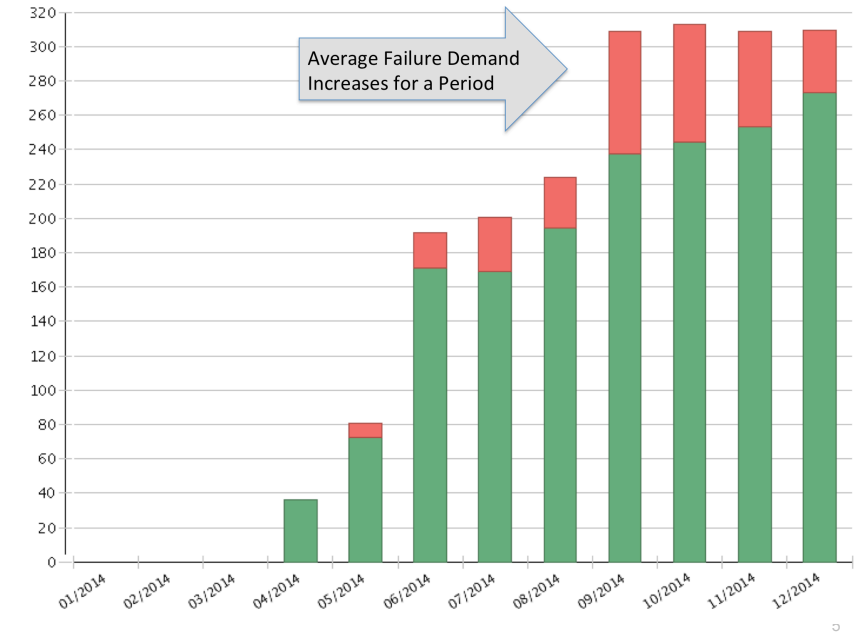
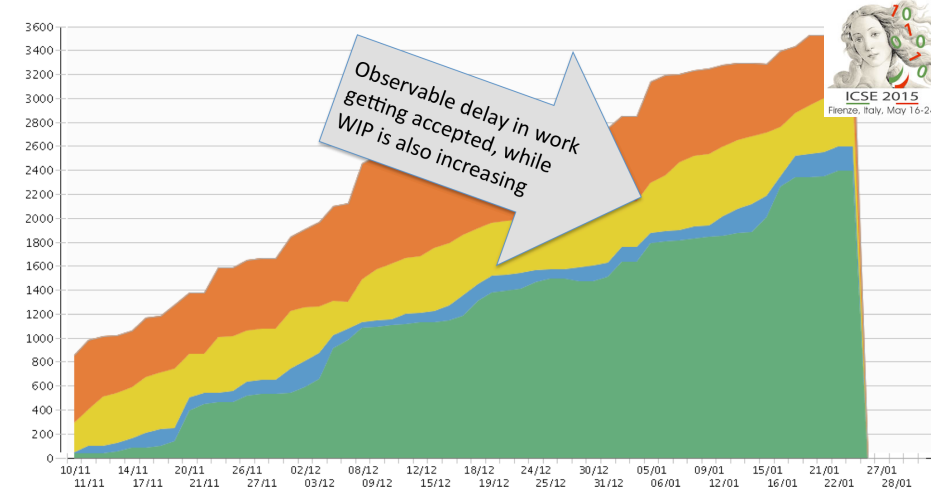
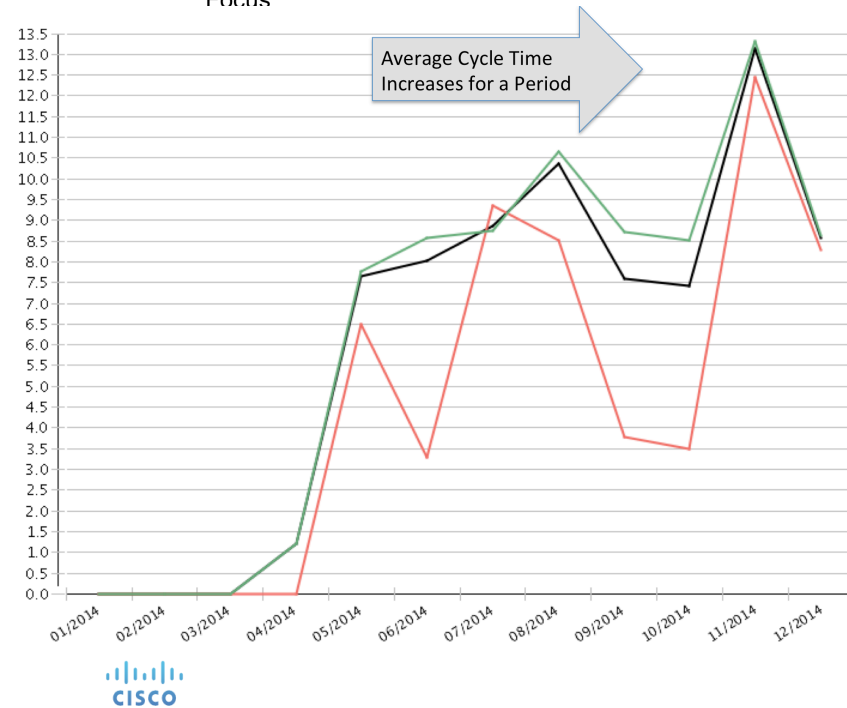


- 1 Computer-controlled sensors in road detect increased traffic and feed data back to control centre
 - 2 Mandatory speed limit imposed to protect queuing traffic and smooth flows
 - 3 Information signs warn or advise drivers of hazards/lane use
 - 4 Overhead signal indicates when hard shoulder is open to traffic
 - 5 Hard shoulder to be used for travel between consecutive junctions
 - 6 Broken-down vehicles use emergency refuge areas to ensure hard shoulder remains clear
 - 7 Controllers use CCTV and sensors to monitor hard shoulder for obstructions/debris and can manually override ATM system
- * Other safety improvements include highly visible roadside emergency phones (8) located behind safety barriers, and lighting (9) at more frequent intervals along entire stretch of motorway



Org Focus
Architecture Requirements Focus

Workflow States at each level



Primary Issues

- Naïve approaches to agility discount the value of architecture
 - The impact of this scales non-linearly with the size of the system and the organization
- “Definition of Ready” not met
- Recognizing “non-functional requirements” / *ilities
- Maintaining alignment across multiple products and solutions
- Follow-through: From definition to implementation, and incorporating new learning



Other Issues

- Dependencies!
- Architecture in isolation
- Architecture Debt
- Role of types of architect
- Articulating architecture as an investment
- Balancing emergent and upfront architecture
 - BDUF → EDUF
- Balancing centralized and distributed decision making

Questions for Further Discussion

What specific architecture issues can be addressed by **focusing on Flow**, and on **removing impediments**?

What other **metrics** serve as **leading indicators**?

What does this reveal about hidden **architecture debt**?

How can this help with the **balance between alignment and autonomy** in large systems and organizations?



@ken_power



<http://www.linkedin.com/in/kenpower>



ken.power@gmail.com



<http://kenpower.ie/>

Second International Workshop on Software
Architecture and Metrics



TOMORROW starts here.

A Metric-Based Approach to Managing Architecture-Related Impediments in Product Development Flow
An Industry Case Study from Cisco

Ken Power
Cisco Systems, Inc.
Galway, Ireland
ken.power@gmail.com

Kieran Conboy
National University of Ireland, Galway
Galway, Ireland
kieran.conboy@nuigalway.ie