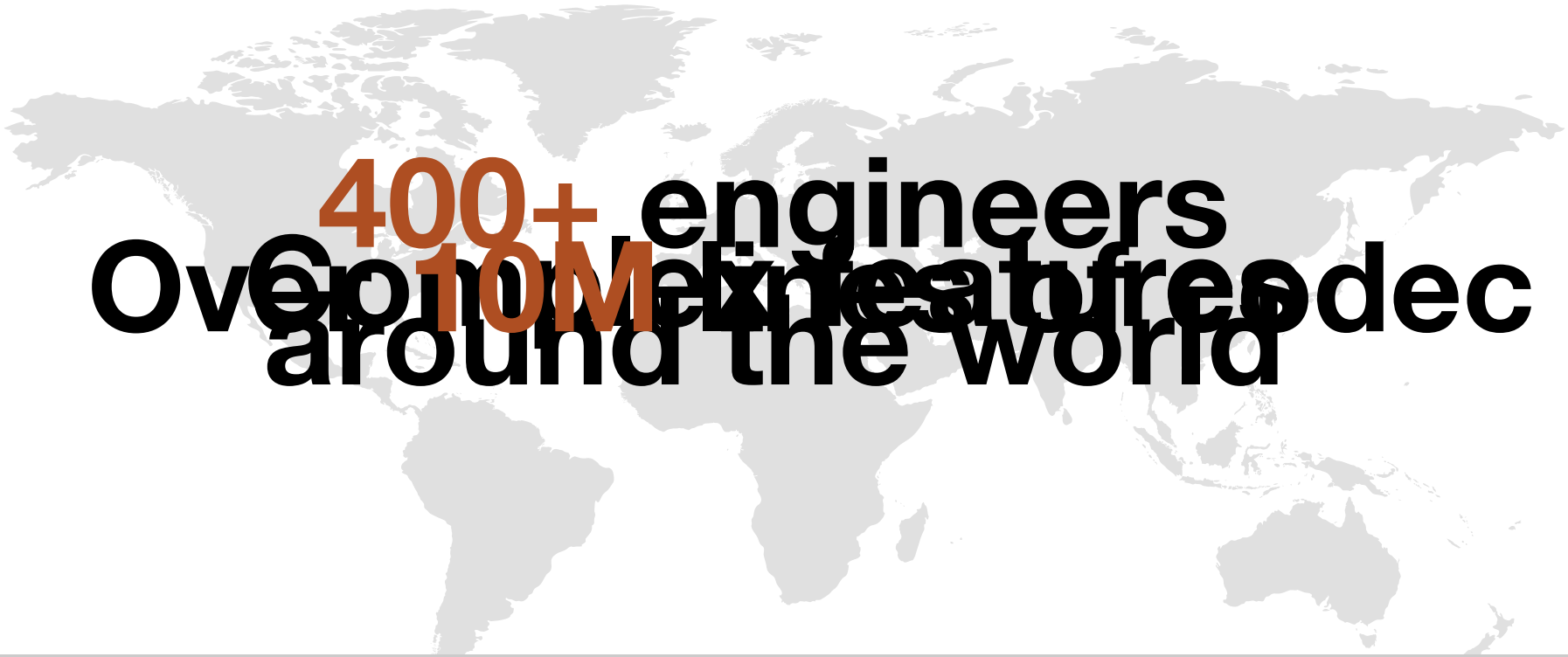
FLOWCON
CONTINUOUS DESIGN DELIVERY DATA

A Practical Approach to Large Scale Agile Development

Gary Gruver
November 1, 2013

4+ Year Large-Scale Agile Journey



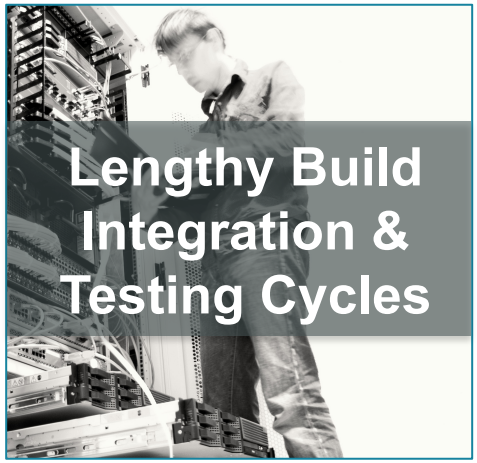
400+ engineers

10M lines of code

Over 10 Mexico freedec around the world

- High-end LaserJet printers and MFPs
- Embedded SW & FW
- Digital Sending and HP open Extensibility Platform

State of the Development Process: 2008



- 6 weeks + to get through a complete testing cycle (mainly manual)
- Build integration taking 15-20% of resources a week to get fixes to main
- Manual testing a key driver and constraint for adding products

- Ongoing customer issues with consistency and lack of features
- Marketing had essentially given up asking for FW innovations

State of the Development Process: 2008

- Development costs growing 2.5X from 2004-2008 and the business was still constrained
- Up to 10 different branches (driven by each product release window) in MFP
- CPE driving millions/year in CPE investments

- **80-90% of resources** just porting existing FW to new products and qualifying
- Unable to add new products to the plans due to lack of FW resources
- 20% of resources developing plans that quickly became obsolete

Firmware Development Transformation

Consistent Dev Environment

Integrated Tools

Organizational Change Management

Agile Development with Mini Milestones (Sprints)

Architected for product variability

Fully automated unit and system test

Continuous integration and test system

One branch for all products including CPE

Breakthrough Capacity for Development

New Customer Capabilities

Defect Fixes

FutureSmart FW Large Scale Agile Development Engine

- 400+ developers
- 10+M LOC
- 75,000-100,000 LOC turmoil
- 100-150 Commits
- 10-15 builds /day
- 15,000 hours/day of testing (90% pass rate)

Cycletime Driver Improvements

2008	2011
Build Bosses **1 Week**	Continuous Integration **3hrs**
Number of Builds **1-2**	Continuous Integration **10-15/Day**
Feedback on Main **1 Commit/Day**	Autorevert **~100 Commits/Day**
Full Manual Registration **6 Weeks**	Auto Regression Testing **24 Hrs**

Development Cost Driver Improvements

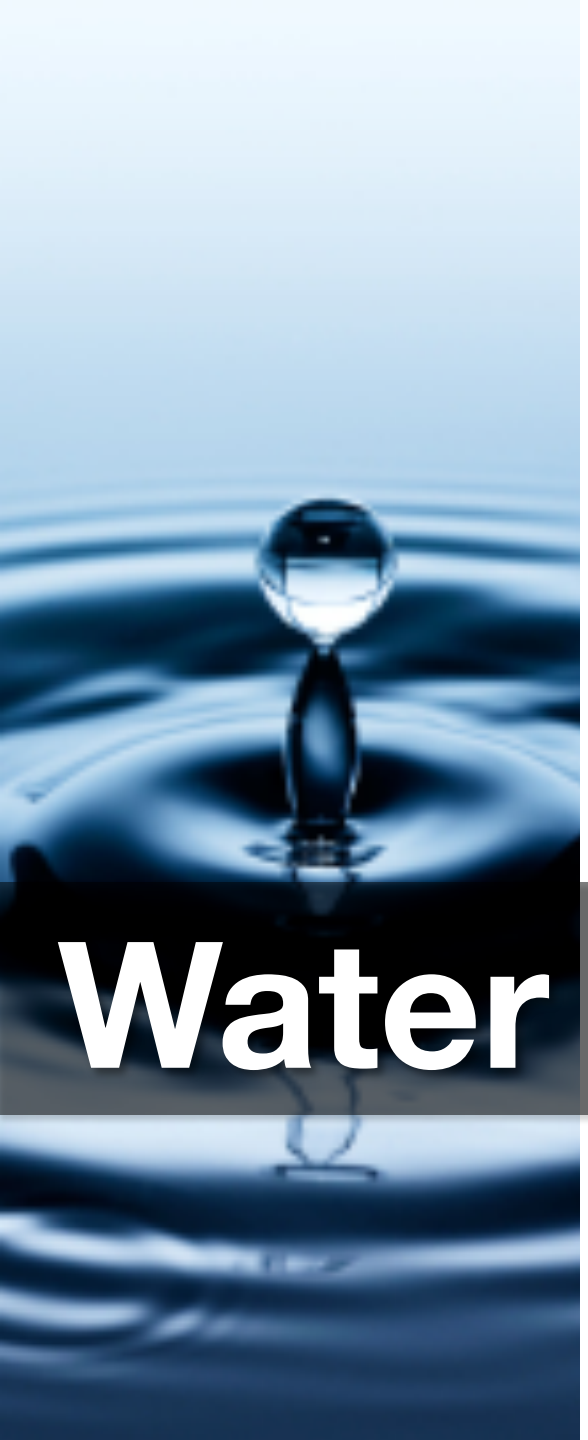
2008	2011
Code Integration **10%**	Continuous Integration **2%**
Detailed Planning **20%**	Agile Planning **5%**
Porting Code **25%**	One Main Branch **15%**
Current Product Support **25%**	One Branch CPE **5%**
Manual Testing **15%**	Most Testing Automated **5%**
Capacity for Innovation **~5%**	***Capacity for Innovation*** **~40%**

State of the art FW development model

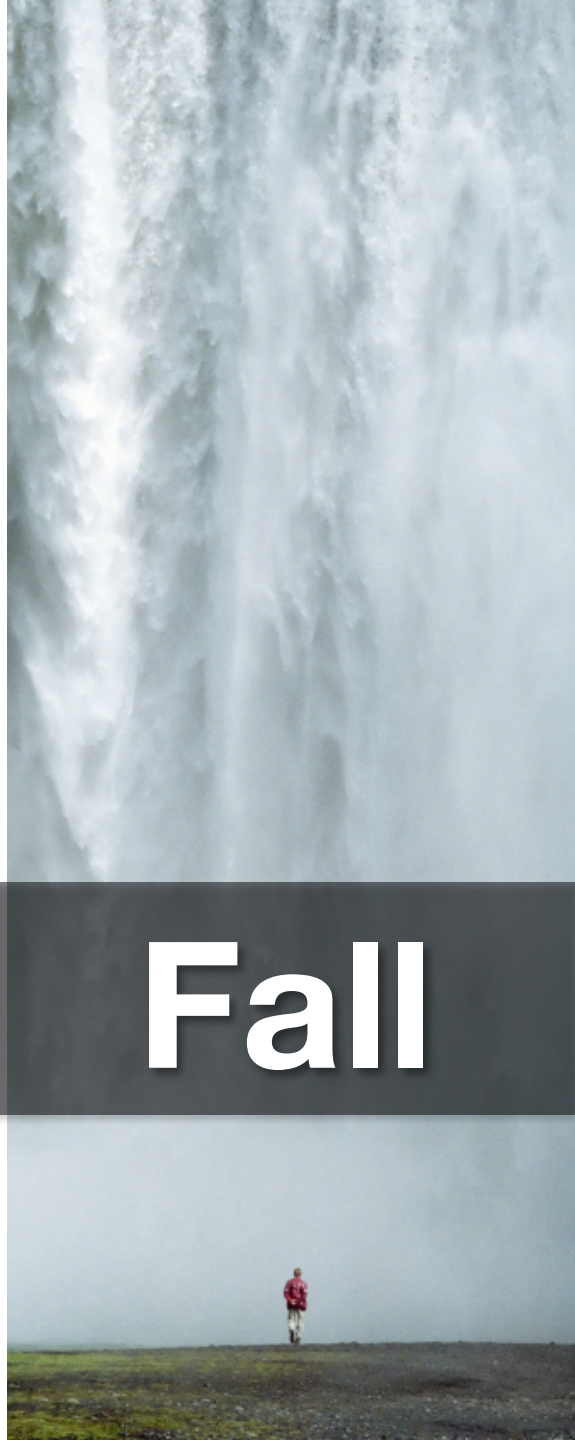
2008	2011
Costs out of control	~70% reduction in FW development cost per program
Couldn't add resources fast enough	50% reduction in FW headcount
Lengthy build, integration and testing cycles	Cont. integration, daily automated regression
Products lagging the competition	***Vintage chart unleashed and capacity for innovation***


Making an Enterprise Agile
vs.
Enabling Small Agile Teams in the Enterprise


Scrum ≠ Agile


Water


Scrum


Fall

Improvements Best Driven at the Enterprise Level

- ✔ Business Objectives/ Priorities
- ✔ Enterprise Level Continuous Improvement
- ✔ CI/CD and test automation infrastructure
- ✔ Planning Process

Business Objectives (Don't "Do Agile")

Either automate, eliminate, or engineer out the drivers that aren't key to the value prop

Interative Approach to Agile Management

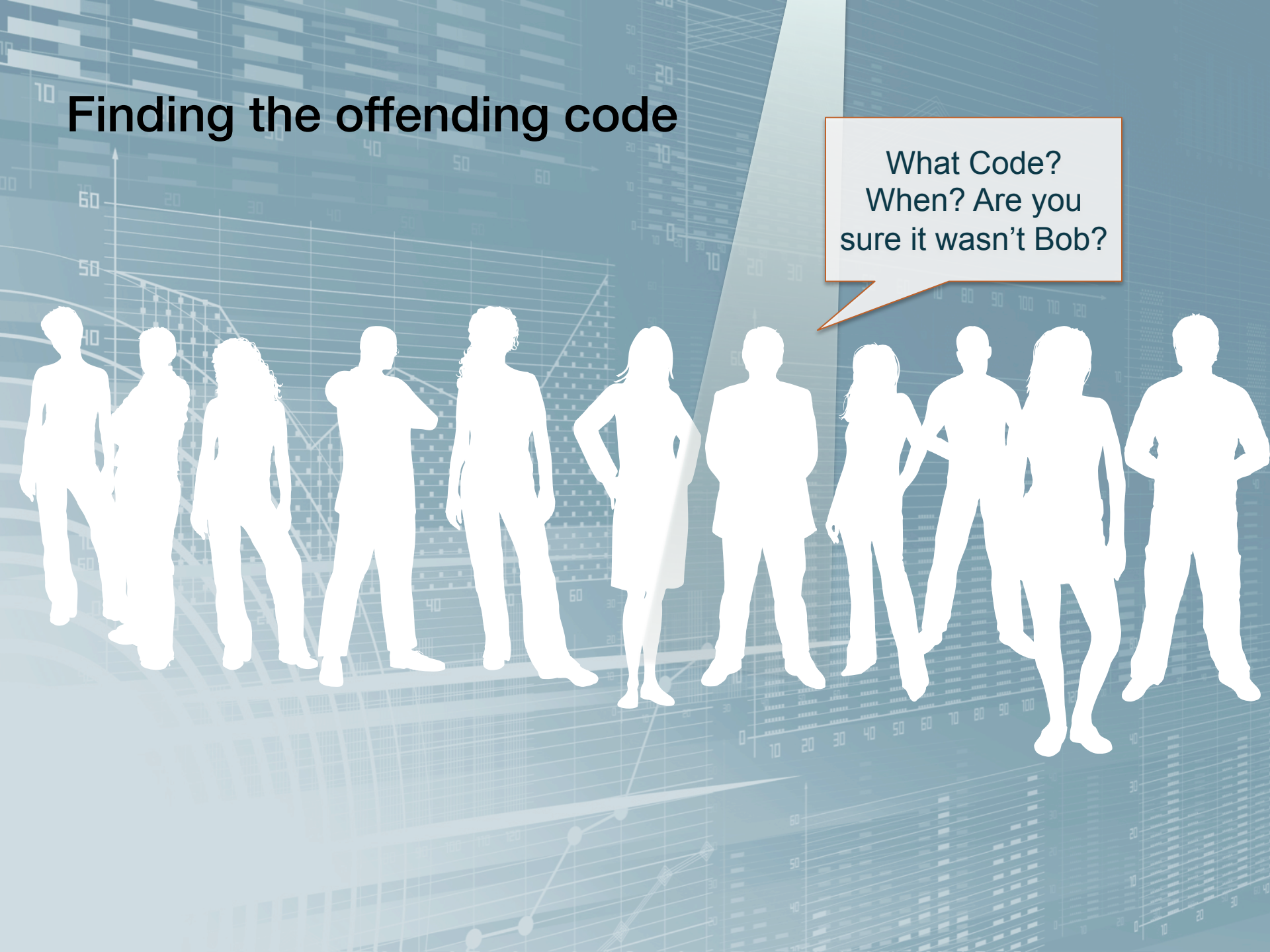
Having real time metrics is essential for the speed of agile & aligning the org.
But don't manage by metrics.

Use the metrics to understand where to have conversations about what is not getting done.

MM30 Objectives

Rank	Theme	Exit Criteria
0	Quality threshold	- P1 open < 1wk - CAT 100% pass - L2 24hr response - Tests for CAT escapes
1	Quarterly Bit Release	A) Final P1 change requests fixed 2 remaining. B) Duration error rate per 10K: 0.3 (sim), 0.35 (emul), 0.4 (product)
2	CE stability and test coverage (PTO3)	A) L2/L3/L4 CAT 100% passing w/ proper coverage (3 superbundles /wk) B) All L2 pillars 98% pass – w/ coverage for high-value PTO1-PTO3 reqts C) L4emu test pillars in place – LLFW, copy/PDL, PrintDevice D) L3 CAT in place with at least L4 CAT equivalence E) L4 test coverage for all PTO1-PTO3 reqts F) Duplicate L4 tests to new products – 100% exec (no DS – okay)
3	PTO4 dependencies and key features	A) Calibration dependencies B) Print for an hour at speed to finisher with stapling C) Copy for an hour at speed 35ppm (40ppm is at speed) D) Enter/exit powersave Approved to push out to MM31 E) Falcon test suite execution Emulator still needs FIM support F) Automated FIM – no bash prompt Approved to push to MM31 G) RUI/CTF support for 4-line display H) Send to Folder, 3rd-party SW avail for Send to Email
4	Build for next-gen products	A) Build single ARM system Feasibility proven. 2 DLL's to re-compile. B) High-level analysis of performance on ARM Lowered priority.
5	Fleet Integration plan	Align on content for "slivers" of end-to-end agile test. Overall plan in place. Need sliver details or will just deliver same as to PTO's.

Finding the offending code



CI/CD and Test Infrastructure

How much of the system do you put together how often and in what order?

How do you build up the system?

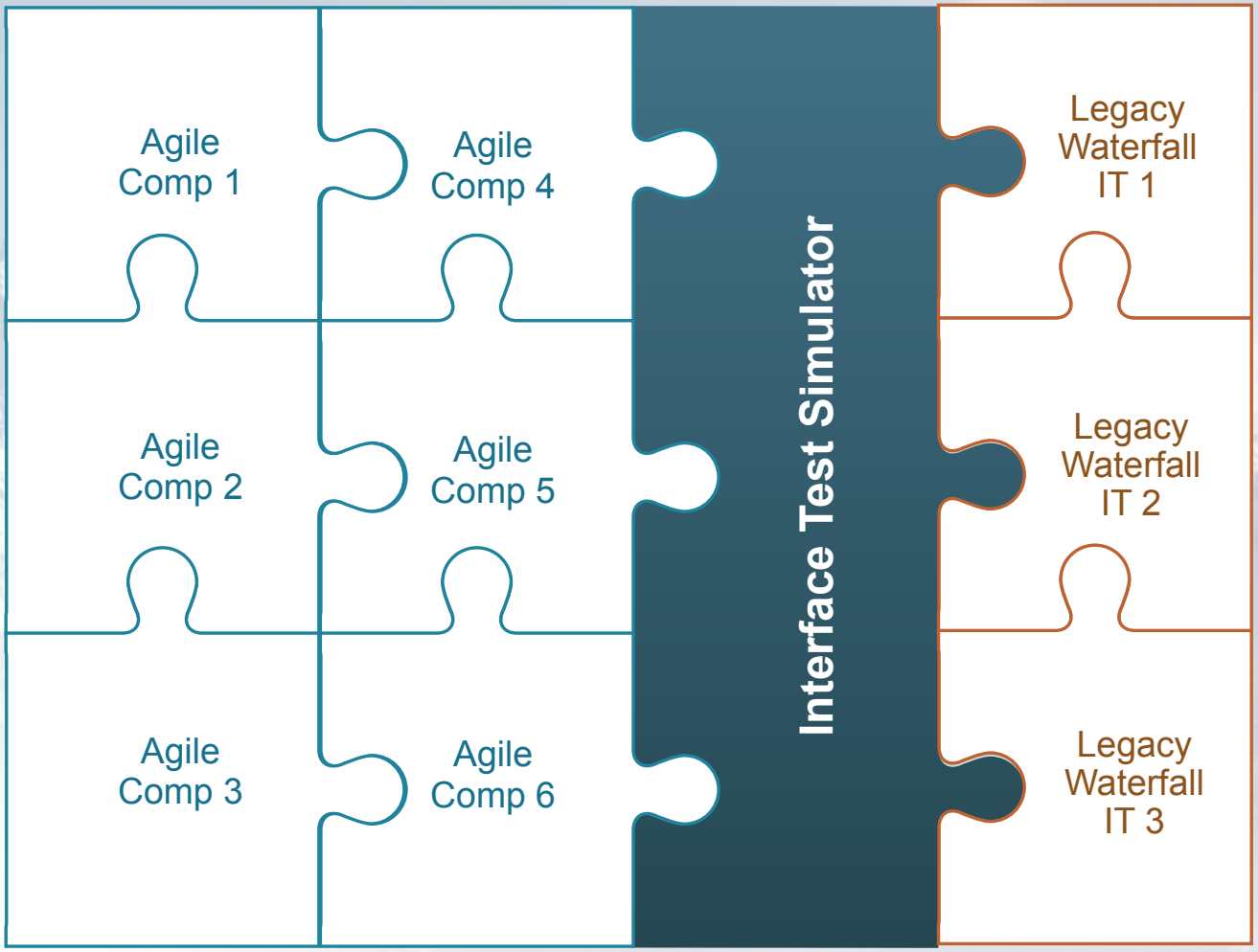
Where do you create test harnesses/simulators/ emulators?

Automated testing is as hard or harder than writing good code.

How do you create frameworks that improve stability and productivity?

Where do you turns builds red and stop the train versus logging defects and tracking passing rates?

Building up a Large SW System

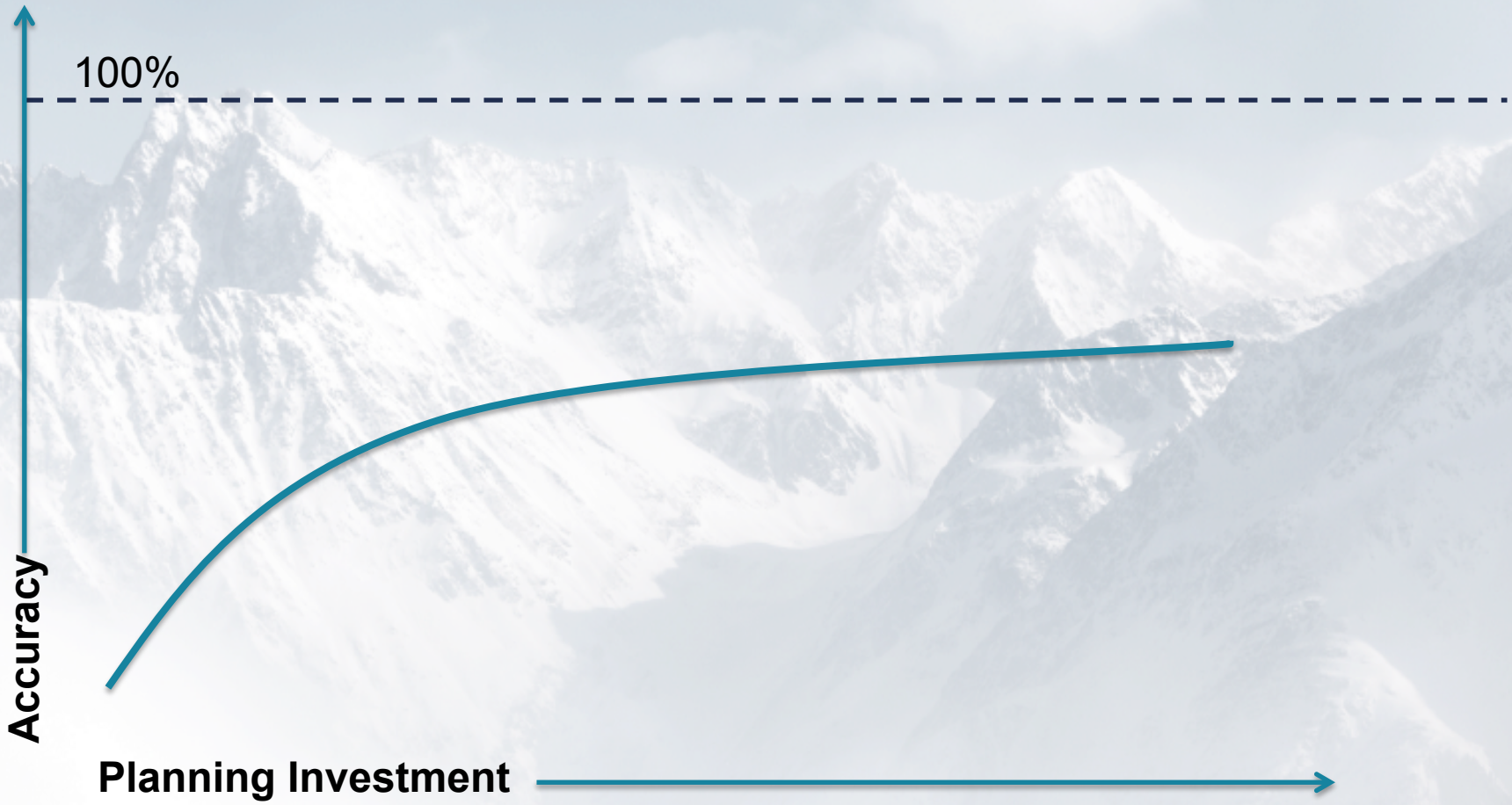


One of the biggest challenges with Agile Planning at the enterprise level is getting the organization to accept the uncertainty in SW development and appreciate the flexibility and opportunity.

Long Term Predictability for SW Schedules

Do we really need the predictability of our current planning processes?

Are our current planning processes really that accurate?



Spr11 1-N High-level Risk/ Resource Analysis

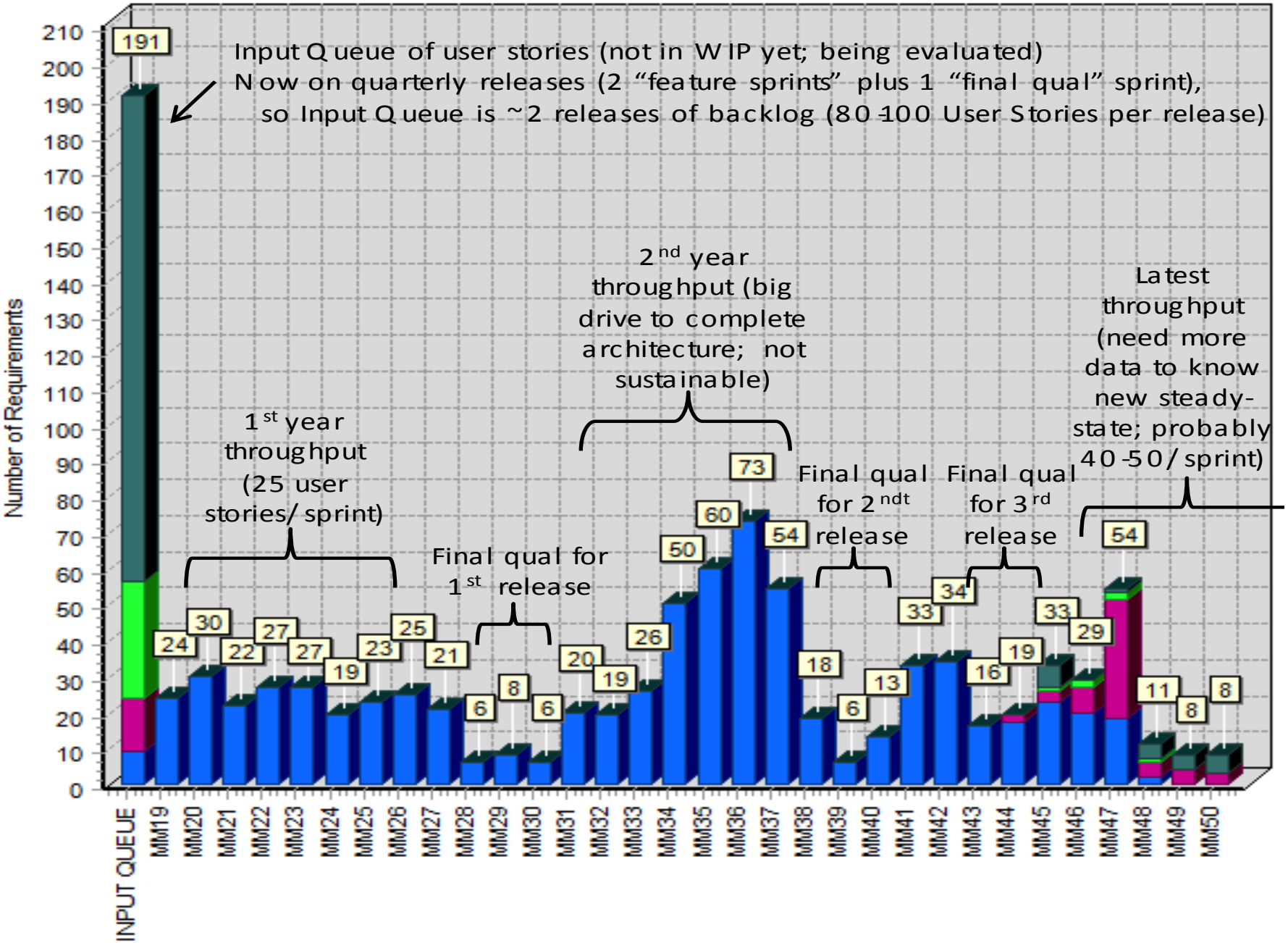
Rank	Initiative	High-Level Estimate – FW Engineering Months												
		Component 1 (25-30)	Component 2 (20-25)	Component 3 (30-40)	Component 4 (30-40)	Component 5 (20-30)	Component 6 (20-30)	Component 7 (20-30)	Component 8 (15-25)	Component 10 (40-50)	Component 11 (20-30)	Component 12 (20-30)	Other teams	TOTAL
1	Initiative A			21			5	3		1				30
2	Initiative B	3							4				17	24
3	Initiative C		5							2	1	1		9
4	Initiative D							10		2	2	2		16
5	Initiative E					20						3	5	28
6	Initiative F	23							5	6			2	36
7	Initiative G									2				2
8	Initiative H											5		5
9	Initiative I												3	3
10	Initiative J		20	27			17			39	17	21	9	[illegible]
11	Initiative K			3	30		3		3	14			[illegible]	
12	Initiative L									2				
13	Initiative M	3						10		6	6	6		[illegible]
		29	25	51	30	20	25	23	12	74	26	38	59	401

Limit long-term commitments to

50-60%

Capacity

FutureSmart Firmware User Stories per Sprint



Getting Mgmt/Mktg Buy-in to Agile Planning

"FW will still commit to basic new product support one year ahead"

Means prioritizing "product turn-on and delivery/qualification" ahead of new features

"You get to decide what we work on first"

Establish a "1-N feature request list" and the combined marketing teams decide the order

"You will get 20% more features this way"

Easy to explain the 20% of resources previously used to estimate

"We'll actually listen to your last-minute requests"

Just put it at the top of the list, ahead of all the other "input queue" features

A Practical Approach to Large Scale Agile Development

E-mail: gbgruver@gmail.com
Blog: largescaleagile.com
Twitter: @GRUVERGary