



Applying TOC to Services

*Cases and lessons learned form IT, public
and health service organizations*

Hans Steenpoorte, TOC Resultants



Hans Steenpoorte

Co-owner of TOC Resultants and Critical Task Manager

TOC implementor in service organizations; government, professional services, IT and healthcare. All implementations are aimed at delivering *more* services *faster*, with the same people and resources.

Hans worked 10 years in the steel and aluminum industry. Since 2000, he moved to professional services and in 2005 founded TOC Resultants with Michel Stijlen. Since then, they have been implementing TOC solutions in services, and adapting them if and where necessary.

In 2009, Hans co-founded Critical Task Manager (CTM), TOC-based software for service organizations that want to deliver repetitive and/or unique services faster and more reliably.

Hans lives in Amersfoort, The Netherlands with his wife, 2 daughters and 2 guinea pigs



www.toc-resultants.com

info@toc-resultants.com

See also:

www.criticaltaskmanager.nl/





Agenda

1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital labs
 3. Municipalities
4. General lessons learnt



Why services?

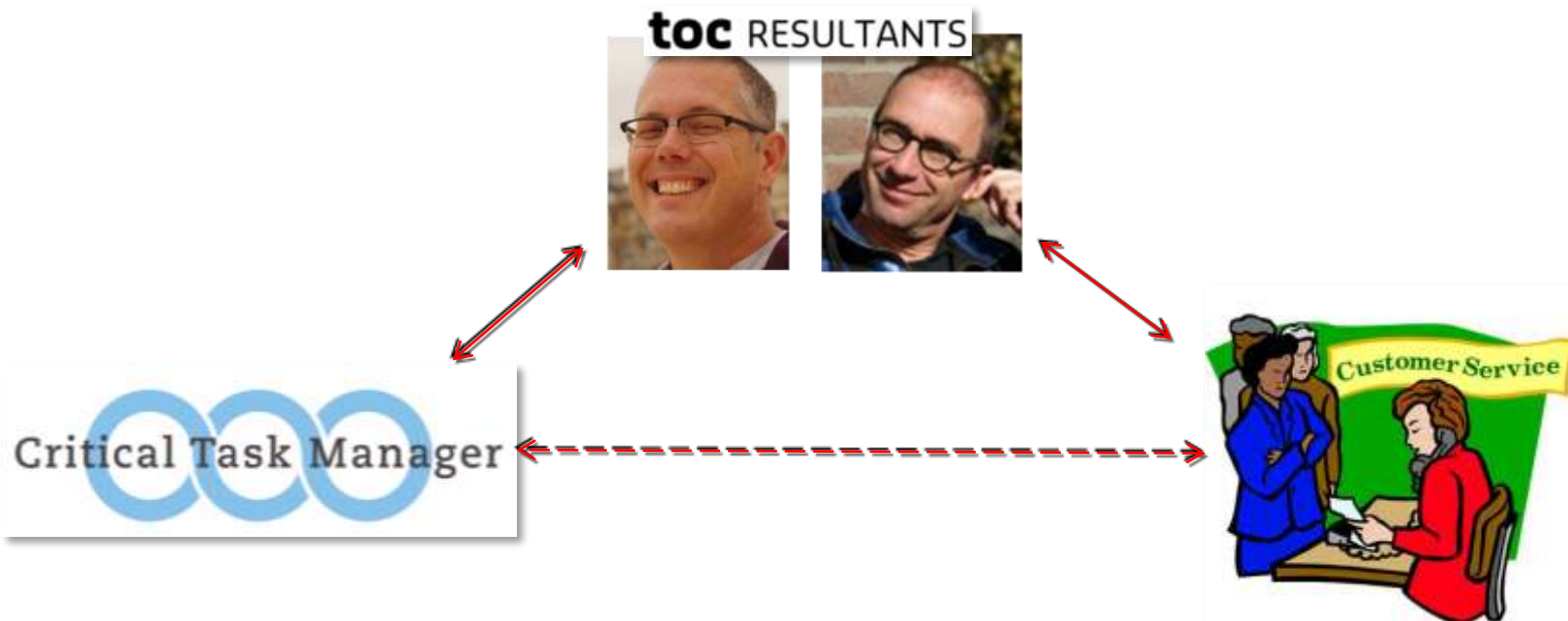
- Isn't that incredibly boring?
 - Production and *professional* project management are consultants' battle ground
 - Services are growing business in W-Europe
 - Much less expertise and competition in services
 - Less dogmatism (6 Σ vs. Lean vs. TOC vs. ?)
 - ***One eyed king in the land of the blind***





Our proposition

- Our expertise “Operations Management of Services”
 - How to plan, execute and improve service delivery so that **more** services are delivered **faster** and more **reliably** with the **same** people/resources and **without compromising quality and/or working conditions**





About services

- Public services (f.i. municipalities)
- Professional services (f.i. lawyers)
- IT services (in- and external)
- Healthcare services
- Banking/insurance
- Etc

How do they compare to production?

Characteristic	Production	Services
Routing & semi's	Well defined	Often poorly/not defined
WIP	Physical	Intangible (or the customer/client!)
Resources	Machines	People (and IT systems)
TT as % of LT	Low	Low, but frequently 'hard to compress/control' LT outside the primary system

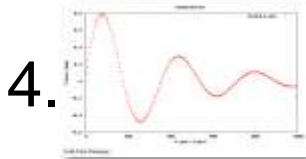


Agenda

1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital labs
 3. Municipalities
4. General lessons learnt



6 solution elements (injections)



5.

Dynamics			
Operational	OC	Begin	End
revenue	cost	revenue	revenue
1000	400	1000	1000
1000	400	1000	1000
1000	400	1000	1000
1000	400	1000	1000



Injection	Benefit
Dramatically reduce WIP	Dramatically decrease LT
(Re-)define/design the process	Less LT for customer and less TT for the professionals
Uniform task priorities (BM)	High DDP
Work in teams	Less peaks & troughs (aggregation)
Management reports & process	Transparency and grip of operational performance
Identify sources & causes of delay	Continuous improvement



About injections # 1 & 2

- Partly IO (instead of injection)

What do we do? “Products/Services Catalogue (PSC)”:

1. Define products/services; predominantly in- and output
2. Define “*ambitious but realistic target lead times*” (buffers)
 - 1 to \approx max 4
3. Define responsible skill groups

Next, we split the WIP in

- A. Services that are still within the (new) TLT’s (\approx 50%)
- B. Services that are already over the TLT’s (\approx 50%)

.... and we split the organization in 2 groups

- Going concern (\approx 80% of capacity); takes care of A
- Intermediate sprint (\approx 20% of capacity); takes care of B



Intermediate sprint?

- (Relatively) senior multi-disciplinary group of people ***physically (!) put together***



Objective:

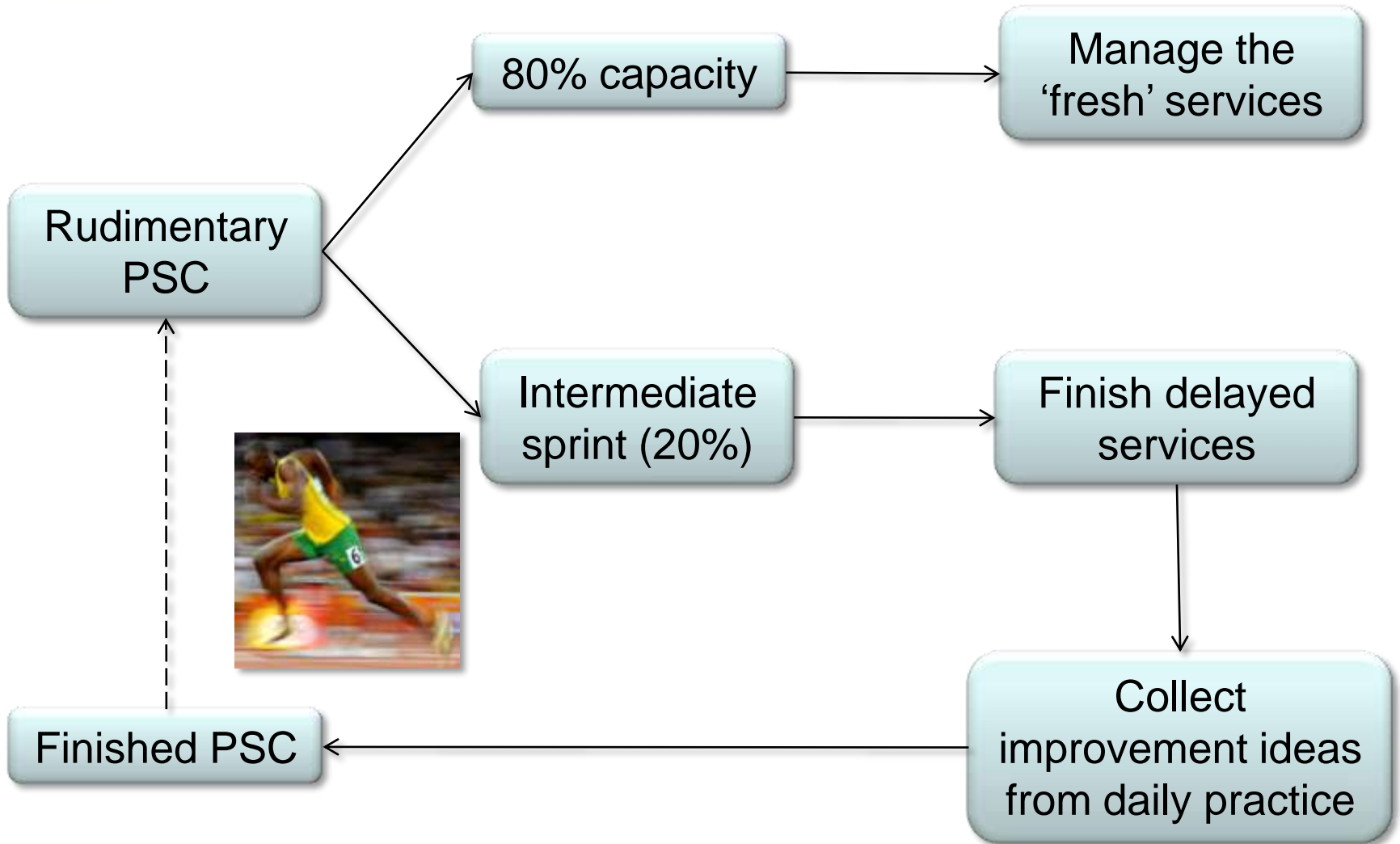
1. Rapidly reduce WIP by finishing overdue services (starting from the oldest)
 - ✓ No fancy BM tools at this point; just lists on the wall
 2. Collect ideas during daily stand-ups about how to simplify and uniform the process(es)
- Input for finishing the Products/Services Catalogue (PSC)”

Question: Can the other 80% of the capacity keep up with 100% of the (new) inflow?

- Yes: Delayed services are often complex and cause extra load



So conceptually





Agenda

1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital labs
 3. Municipalities
4. General lessons learnt



IT Services

- Objective: Create & maintain an IT solution environment that allows people to work effective- and efficiently
- Often internal service provider

Services (ITIL)



- Incidents
- Changes (in soft- a/o hardware configuration)
 - Standard and non-standard
- New (software) releases
- Problems (causes of multiple incidents)
- System services (back ups, restores, performance monitoring etc)
- Project tasks
- Advice/consulting



Findings



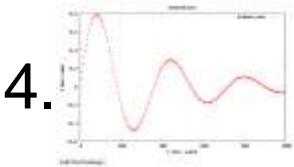
1. Intermediate sprint of only 1-2 weeks!



2. Mainly defining in- and output and 'translating' into check lists



3. Integrate with existing call registration system



5.

	Degree of	Impact
...
...
...
...
...



4-6: Challenge



Challenge: teams

- (Partly) because of ongoing specialization, IT service management is increasingly individualistic
 - Frustrates aggregation and process improvement

So

- We form semi-homogeneous “skill groups”
- Appoint a foreman per group
- Implement clear management processes & reports:
 - **Daily:** Foreman checks that calls are processed according to bufferstatus (“dispatching”)
 - **Weekly:** Foremen meet with IT Manager to monitor production, LT and DDP and solve “**Top 20**” delayed calls
 - **Monthly:** Analyse and implement sources & causes of delay





Results (3 implementations)

- Production: 0-20% decrease!
- LT: More than cut in half
- DDP: From $< 50\%$ to $> 90\%$
- Less hectic working environment
- More structured improvement process(es)





Agenda

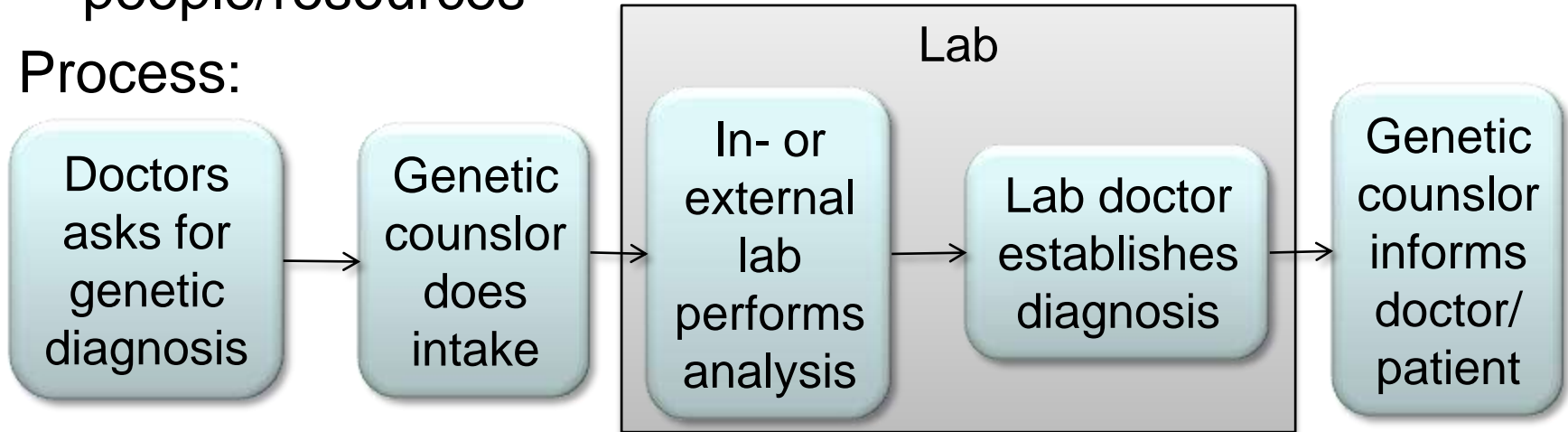
1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital labs
 3. Municipalit
4. Genaral lessons learnt



Academic genetic diagnostics labs

- Objective: Correctly perform **ever increasing** numbers of genetic analyses faster en more reliably with \pm current people/resources


Process:



- LT's of 1 month to several years (no typo)
- WIP (patient files) everywhere
- No visibility/grip on logistical performance
- Stressed and frustrated work force



Results (3 implementations)

- Output increased by 10-20% (vs. 2011) with same resources
- LT's dropped from 11 to appr. 3 months
- Access time decreased from months to weeks/days
- In the labs the work load increase is absorbed quite easily
- Workload at counselors was/is quite high 
 - Difficult to decrease, mainly because most counselors find it very hard to agree on standard generic processes that facilitate better support and delegation



Agenda

1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital lab
 3. Municipality
4. General lessons learnt



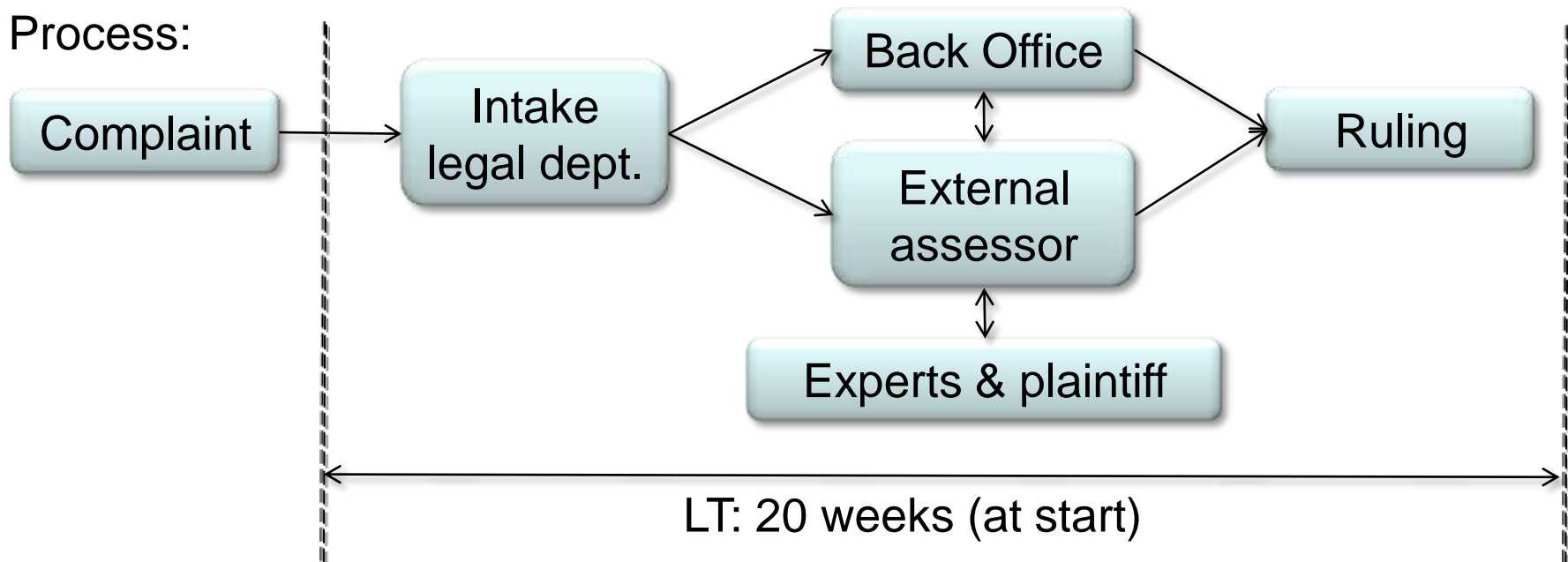
Municipal social services

- Objective: To correctly and timely process resident's objections to municipal rulings
- Challenge: When LT > 6 weeks, the plaintiff is equated automatically and the town council is not amused

➤ Enter Mr. Wolf

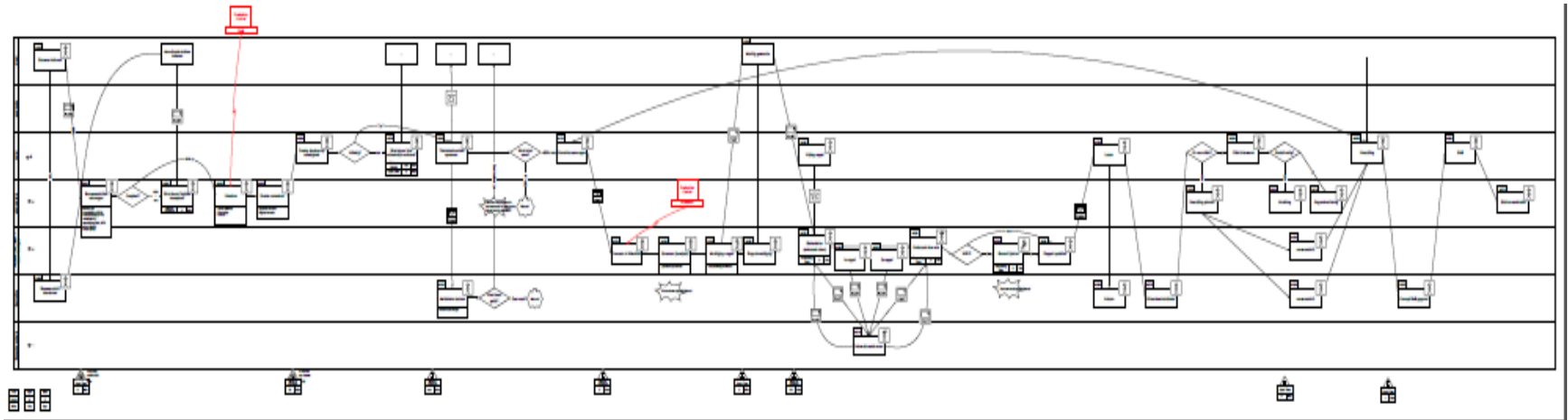


Process:





“Shocking” findings ;-)



Conclusions:

- Realization that current process is DOA
- Lots of unnecessary
 - Steps &
 - Sequentiality
- In 2 weeks intermediate sprint we gathered 40 improvement ideas



(Intermediate) results

- LT decreased from 20 to 10 weeks in 2 months. No doubt it will come down to 4 weeks with the new process
- Productivity of lawyers in the Sprint Team is 3 X higher than normal (5 => 15 complaints/ week)
- Multi-disciplinary proximity appears to have many benefits
- So what to do ?
 - Keep division of labour (“skill groups”) to facilitate aggregation?
 - Create 1 or more (?) multi-disciplinary teams to speed up interaction?



Agenda

1. About services
2. Generic solution
3. Cases
 1. IT management
 2. Hospital lab
 3. Municipality
4. General lessons learnt



1. We found another hero

- 1st, this guy ...



then



and now



WIP (#units) = Average LT (in days)

Output/day

- Since WIP in services is often invisible and not an investment, the levels and problems caused by it are often huge
- Little's Law is easy to understand & basis for management reports:
 - Input, Output, WIP, LT and DDP



2. Intermediate sprint

- We found that even a “*general direction of flow*” (Eli G) is sometimes hard to discern
 - It is useful/necessary to (re)define the products and processes relatively early on in the implementation, rather than to wait for BM stats
- Intermediate sprint is an effective way to ...
 - Reduce WIP fast (whilst continuing service to current and new clients)
 - Create new insights that are used to (re) design/define the products/services and process
- Creates immediate buy-in, because people are often frustrated with (part of) the process for years
- On top, machines don’t appreciate TT reduction. People do!



3. How to deal with project-like services?

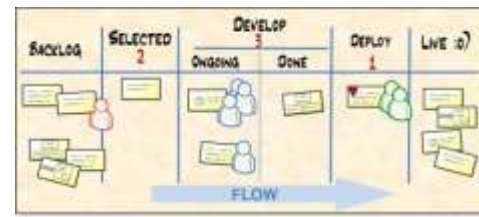
- > 80% of services can be managed with (a derivative) of S-DBR; clean and simple
- Some however, really are more project-like in having IO's, parallel tasks, hard-to-compress LT's and unpredictable flow
- Their tasks often run through the same resources as the S-DBR tasks
- We find CCPM to be too complex and rigid to integrate into 1 solution for delivering services

Solutions (?)

- Critical Task Manager?
- Kanban (David Anderson)?

Hmmmmm

- Are we alone?





So, what's our ambitious target?

**LET'S MAKE TOC THE MAIN
WAY OF MANAGING SERVICE
ORGANIZATIONS**