The Empirical Process

- at the very core of agile
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...
An everyday empirical process...

Do I jump in...?

Me

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
An everyday empirical process...

Schematic:

Do I jump in...?
An everyday empirical process...

Do I jump in...?

Desired

Schematic:
An everyday empirical process...

Do I jump in...?

Schematic:

Desired
An everyday empirical process...

Desired

Controller

Do I jump in...?

Schematic:

Me

Desired

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?

Schematic:

Desired

Controller

Me

Wednesday, August 10, 2011
An everyday empirical process...

Do I jump in...?

Schematic:

Desired

Controller

System input

Me
An everyday empirical process...

Schematic:

Controller

Desired

Me

Do I jump in...?

System input

Desired

System input

Controller
An everyday empirical process...

**Schematic:**

Do I jump in...?

Controller

Desired

System

Me

Controller

System input

Me

System input

Desired

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?

Schematic:

Controller

System

Desired

System input

Me

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
An everyday empirical process...

Schematic:

Do I jump in...?
An everyday empirical process...

Do I jump in...?

Schematic:

Feed-forward or Open-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?
Actual == Desired?

Schematic:

Feed-forward or Open-loop system
An everyday empirical process...

Do I jump in...?
Actual==Desired?

Schematic:

Feed-forward or Open-loop system
An everyday empirical process...

Do I jump in...?
Actual==Desired?

Controller
System

System
input

Controller

Desired

Actual

Desired

Temp

Controller
System

Feed-forward or Open-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?
Actual==Desired ?

Controller
System

Schematic:

Feed-forward or Open-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...? 
Actual==Desired ?

Controller

Schematic:

Feed-forward or Open-loop system

Daniel Brolund – @danielbrolund – daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?
Actual==Desired ?

Schematic:

Feed-forward or Open-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

Do I jump in...?  
*Actual == Desired?*

Schematic:

- Controller
- System

*Feed-forward or Open-loop system*

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
An everyday empirical process...

**Do I jump in...?**

Actual==Desired ?

**Schematic:**

Feed-forward or Open-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
What is the control error?
What is the control error?

Temp

Time
What is the control error?

![Graph showing the relationship between temperature, desired, and time.](image)
What is the control error?

Temp

Desired

Actual

Time
What is the control error?

Temp

Desired

Actual

Time
What is the control error?

"Error surface"
What is the control error?

Control Error = \int |\text{Desired} - \text{Actual}|
What is the control error?

Control Error = \int |\text{Desired} - \text{Actual}|
What is the control error?

Control Error = $\int |\text{Desired} - \text{Actual}|$

"Error surface"
What is the control error?

Control Error = \int |\text{Desired} - \text{Actual}|
The "simplistic" solution

![Diagram showing a control system with desired input, controller, system, and actual output connections.]

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The "simplistic" solution

Controller = (System)^{-1}
The "simplistic" solution

Controller = (System)\(^{-1}\)

\[ \Rightarrow \text{Desired} = \text{Actual} \]
The "simplistic" solution

Controller \((\text{System})^{-1}\) 
\[\Rightarrow \text{Desired} = \text{Actual}\]

Requires:
- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input
The "simplistic" solution

Controller = (System)^-1

⇒ Desired = Actual

Requires:
- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input
The "simplistic" solution

\[ \text{Controller}=\left(\text{System}\right)^{-1} \]

\[ \Rightarrow \text{Desired} = \text{Actual} \]

Requires:

- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input
The "simplistic" solution

\[\text{Controller}=(\text{System})^{-1}\]

\[\Rightarrow \text{Desired} = \text{Actual}\]

**Requires:**
- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The "simplistic" solution

Controller

Controller=(System)^{-1}
⇒ Desired = Actual

Requires:
- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input
The "simplistic" solution

Controller \(= (System)^{-1}\)

\[\Rightarrow \text{Desired} = \text{Actual}\]

Won't work in reality

Requires:
- detailed knowledge of System, Desired and hence Actual
- that such a Controller exists and can create such System input
- that the System can handle System input
Adding feedback!

Desired

Controller

System input

System

Actual

Me

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Desired | Controller | System | Actual
------- | ---------- | ------ | ------

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Wednesday, August 10, 2011
Adding feedback!

Schematic:

Desired

Controller

System

Actual

System input

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Desired -> Controller -> System -> Actual

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

- Desired
- Controller
- System
- Actual
- Observed result
- Sensor
- Me
- System input

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Desired

Controller

System

Actual

Observed result

System input

Sensor

Desired

Controller

System

Actual

Observed result

Sensor

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Feed-back or Closed-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Adding feedback!

Schematic:

Feed-back or Closed-loop system

Observed result

Controller

System input

System

Actual

Sensor

Temp

Time

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Feed-back or Closed-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Adding feedback!

Schematic:

Desired

Controller

System

Actual

Controller

System input

Actual

Sensor

Temp

Desired

Actual

Time

Feed-back or Closed-loop system

Wednesday, August 10, 2011

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Adding feedback!

Schematic:

Feed-back or Closed-loop system

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
What is the control error now?
What is the control error now?

<table>
<thead>
<tr>
<th>Temp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desired</td>
</tr>
<tr>
<td>Actual</td>
</tr>
</tbody>
</table>
What is the control error now?
What is the control error now?

\[ \text{Control Error} = \int |\text{Desired - Actual}| \]
What is the control error now?

Control Error = $\int |\text{Desired} - \text{Actual}|$
What is the control error now?

\[ \text{Control Error} = \int |\text{Desired} - \text{Actual}| \]

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
What is the control error now?

Control Error = $\int |\text{Desired} - \text{Actual}|$

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
What is the control error now?

Control Error = $\int |\text{Desired-Actual}|$

bounded!!

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Exercise on feed-forward systems
Exercise on feed-forward systems

Drop-zone
Exercise on feed-forward systems

Drop-zone

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Exercise on feed-forward systems

Robot

Drop-zone
Exercise on feed-forward systems
Exercise on feed-forward systems

Robot

Controller

Drop-zone

↑ ⊤ P D

Wednesday, August 10, 2011

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Exercise on feed-forward systems

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Exercise on feed-forward systems

Tracker counts paper balls in dropzone as a function of time
Exercise on feed-forward systems

Tracker counts paper balls in dropzone as a function of time

Desired  
Measured

Time (s)

Wednesday, August 10, 2011
Exercise on feed-forward systems

Goal: Meet desired nr of paper balls in dropzone (see graph)

Tracker counts paper balls in dropzone as a function of time
Exercise on feed-back systems

- Desired
- Measured

Time (s)

Desired vs. Measured Graph

Robot

Controller

Tracker

Drop-zone

Exercise on feed-back systems

Exercise on feed-back systems

Exercise on feed-back systems

Exercise on feed-back systems
Exercise on feed-back systems

As before, but now only one step at a time!

Time (s)

Desired
Measured

Robot
Tracker
Drop-zone
Controller

Daniel Brolund – @danielbrolund – daniel.brolund@agical.com

Wednesday, August 10, 2011
Exercise on feed-back systems

As before, but now only one step at a time!

Desired
Measured

Time (s)

Exercise on feed-back systems

As before, but now only one step at a time!

Desired
Measured

Time (s)
Exercise on feed-back systems

As before, but now only *one step at a time*!
Exercise on feed-back systems

As before, but now only one step at a time!
Exercise on feed-back systems

As before, but now only one step at a time!
Exercise on feed-back systems

As before, but now only one step at a time!

Exercise on feed-back systems

As before, but now only one step at a time!
Exercise on feed-back systems

As before, but now only one step at a time!

Desired
Measured

Controller
Robot
Drop-zone
Tracker

Time (s)
0 20 40 60 80 100
0 2 4 6

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Exercise on feed-back systems

As before, but now only one step at a time!

Exercise on feed-back systems

As before, but now only one step at a time!
Points of Discussion - groups of 4-6

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Points of Discussion - groups of 4-6

Results in exercises?
Errors?
Points of Discussion - groups of 4-6

Results in exercises? Difference between exercises?
Errors?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?

What are the measurements we make?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?

What are the measurements we make?

Impact of robot precision?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?

Errors?

What are the measurements we make?

What are the unknowns?

Impact of robot precision?
Points of Discussion - groups of 4-6

Results in exercises? Errors?  
Difference between exercises?

What are the measurements we make?  
What are the unknowns?

Impact of robot precision?  

Sensitivity to unknowns & changes? Robustness?
Points of Discussion - groups of 4-6

- Results in exercises?
  - Errors?

- What are the measurements we make?

- Impact of robot precision?

- Sensitivity to unknowns & changes? Robustness?

- Difference between exercises?

- What are the unknowns?

- Iteration length?

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Points of Discussion - groups of 4-6

Results in exercises? Errors?

Difference between exercises?

What are the delays?

What are the unknowns?

What are the measurements we make?

Impact of robot precision?

Iteration length?

Sensitivity to unknowns & changes? Robustness?
Points of Discussion - groups of 4-6

Results in exercises? Errors?

What are the measurements we make?

Impact of robot precision?

Sensitivity to unknowns & changes? Robustness?

Difference between exercises?

What are the delays?

What are the unknowns?

Iteration length?

Impact of iteration length?
Points of Discussion - groups of 4-6

Results in exercises? Errors?

Difference between exercises?

What are the delays?

What are the unknowns?

Iteration length?

Impact of iteration length?

Sensitivity to unknowns & changes? Robustness?

Impact of robot precision?

Similarities with software development?
The impact of delay

Desired

Controller

System

Sensor

System input

Actual

Me

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The impact of delay

Schematic:

- Desired
- Error
- System input
- Observed result
- Actual

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Desired Error
Controller System input System
Actual

Observer result
Sensor

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Desired

Error

Controller

System input

System

Delay

Actual

Observed result

Sensor

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Desired Error

Controller System Delay Actual

System input

Sensor

Observed result

Desired

Time

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The impact of delay

Schematic:

Desired → Error → Controller → System → Delay → Actual

Observed result → Sensor

Temp vs. Time

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Desired → Error → Controller → System → Tap temp → Delay → Actual

Desired

- Error
- Observed result
- Sensor

System input

Delay

System

Sensor

Tap temp

Actual

Temp

Time

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The impact of delay

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The impact of delay

Schematic:

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Delay considerations
Delay considerations

Temp

Desired
Tap temp
Observed result
Delay considerations

\[
\text{delay} > \text{half of required cycletime}
\]
\[
\Rightarrow
\]

\textbf{Guaranteed instability!}

(in practice, instability comes earlier)
Delay considerations

\[ \text{delay} > \text{half of required cycletime} \]
\[ \Rightarrow \quad \text{Guaranteed instability!} \]
(in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.
Delay considerations

$\text{delay} > \text{half of required cycletime}$

$\implies$

**Guaranteed instability!**

(in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.
Delay considerations

\[
\text{delay} > \text{half of required cycletime} \\
\Rightarrow \\
\text{Guaranteed instability!} \\
\text{(in practice, instability comes earlier)}
\]

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Delay considerations

\[ \text{delay} > \text{half of required cycletime} \]

\[ \implies \]

**Guaranteed instability!**

(in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Delay considerations

\[ delay > \text{half of required cycletime} \]

\[ \Rightarrow \]

Guaranteed instability!
(in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!
Delay considerations

\[ \text{delay} > \text{half of required cycletime} \]

\[ \implies \]

\textbf{Guaranteed instability!} (in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Delay considerations

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!

\[ \text{delay} > \text{half of required cycletime} \]

\[ \Rightarrow \]

Guaranteed instability!
(in practice, instability comes earlier)
Delay considerations

\[ \text{delay} > \text{half of required cycletime} \]

\[ \Rightarrow \]

**Guaranteed** instability!
(in practice, instability comes earlier)

It often takes about 5-10 times the delay to tune the system.

Delay position in loop is irrelevant w.r.t. instability!

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Observability and controllability
Observability and controllability

Desired → Error

Controller → System

System input

Sensor ← Observed result

Actual

Observability and controllability

- Error
- Controller
  - System input
  - System
  - Actual
  - Uncontrollable
- Observed result
- Sensor
- Desired
Observability and controllability
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).
The Sensor has to measure a *relevant state* (observability) that the Controller can influence (controllability).
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).
Observability and controllability

The Sensor has to measure a \textit{relevant state} (observability) that the Controller can influence (controllability).

Water temperature
Flow
Acidity
Mineral traces
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Water temperature
Flow
Acidity
Mineral traces
Radioactivity
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Water temperature
Flow
Acidity
Mineral traces
Radioactivity

SW
Observability and controllability

The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Uncontrollable

Controller

System

Actual

Desired

Error

System input

Observed result

Sensor

Unobservable

Water temperature

Flow

Acidity

Mineral traces

Radioactivity

Lines of code

SW

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The Sensor has to **measure** a *relevant state* (observability) that the **Controller** can **influence** (controllability).

---

**Desired** → **Controller** → **System** → **Actual**

- **Error**
- **System input**
- **Observed result**

**Sensor** → **Uncontrollable** → **Unobservable**

**Water temperature**
- **Flow**
- **Acidity**
- **Mineral traces**
- **Radioactivity**

**SW**

- Lines of code
- Unit test results

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Water temperature
Flow
Acidity
Mineral traces
Radioactivity

Lines of code
Unit test results

SW
ROI

SW

Wednesday, August 10, 2011

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Water temperature
Flow
Acidity
Mineral traces
Radioactivity

Lines of code
Unit test results
Acceptance test results

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Observability and controllability

The Sensor has to measure a *relevant state* (observability) that the Controller *can influence* (controllability).

Water temperature
Flow
Acidity
Mineral traces
Radioactivity

SW

- Lines of code
- Unit test results
- ROI
- Acceptance test results
- Nr of checkins

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Observability and controllability

The Sensor has to measure a **relevant state** (observability) that the Controller can influence (controllability).

- **Desired**
- **Error**
- **System input**
- **Actual**

**Uncontrollable**

**Unobservable**

**Water temperature**
**Flow**
**Acidity**
**Mineral traces**
**Radioactivity**

**Lines of code**
- Unit test results
- Acceptance test results
- Request statistics

**SW**

**ROI**
**Nr of checkins**

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The Sensor has to measure a **relevant state** (observability) that the **Controller** can influence (controllability).

### SW
- Lines of code
- Unit test results
- ROI
- Acceptance test results
- Nr of checkins
- Request statistics
- User satisfaction

### Water temperature
- Flow
- Acidity
- Mineral traces
- Radioactivity

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
The Sensor has to measure a relevant state (observability) that the Controller can influence (controllability).

Observability and controllability

Uncontrollable

Desired

Error

Controller

System input

System

Actual

Observed result

Sensor

Unobservable

Water temperature
Flow
Acidity
Mineral traces
Radioactivity

SW

Lines of code
Unit test results
ROI
Acceptance test results
Nr of checkins
Request statistics
Usability
User satisfaction

Wednesday, August 10, 2011
There are many feedback loops at play!
There are many feedback loops at play!
There are many feedback loops at play!

Live user feedback/release
There are many feedback loops at play!

Live user feedback/release  Usability tests
There are many feedback loops at play!

Live user feedback/release  Usability tests
There are many feedback loops at play!

Live user feedback/release  Usability tests  Retrospective
There are many feedback loops at play!

Live user feedback/release  Usability tests

Whiteboard discussions
Retrospective
There are many feedback loops at play!

Live user feedback/release  Usability tests

Daily stand-up
Whiteboard discussions
Retrospective

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
There are many feedback loops at play!

- Daily stand-up
- Whiteboard discussions
- Retrospective
- Live user feedback/release
- Usability tests
There are many feedback loops at play!

ATDD
Daily stand-up
Whiteboard discussions
Retrospective

Live user feedback/release
Usability tests
There are many feedback loops at play!

- TDD
- ATDD
- Daily stand-up
- Whiteboard discussions
- Retrospective
- Live user feedback/release
- Usability tests
There are many feedback loops at play!
There are many feedback loops at play!

Live user feedback/release  Usability tests

Pairing  TDD  ATDD  Daily stand-up

Whiteboard discussions  Retrospective

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
There are many feedback loops at play!
There are many feedback loops at play!

Squiggles

Compilation

Pairing

TDD

ATDD

Daily stand-up

Whiteboard discussions

Retrospective

Live user feedback/release

Usability tests

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
There are many feedback loops at play!

Feedback cycle-time sweet-spots are different due to processing and delays
There are many feedback loops at play!

Feedback cycle-time sweet-spots are different due to *processing and delays*.

Unstable system

Live user feedback/release

Usability tests

Squiggles

Compilation

Pairing

TDD

ATDD

Daily stand-up

Whiteboard discussions

Retrospective

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
There are many feedback loops at play!

Feedback cycle-time sweet-spots are different due to *processing* and *delays*

- too slow ⇒ Unstable system
- too fast ⇒ Controller overloaded or too expensive
There are many feedback loops at play!

Feedback cycle-time sweet-spots are different due to *processing* and *delays*

- Squiggles
- Compilation
- Pairing
- TDD
  - ATDD
  - Daily stand-up
  - Whiteboard discussions
  - Retrospective
- Live user feedback/release
- Usability tests

...AND MANY MORE!

too slow ⇒ Unstable system

too fast ⇒ Controller overloaded or too expensive

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Is feed-forward waste?
Is feed-forward waste?

Desired → Error → Controller → System input → System → Actual

Observed result → Sensor

Wednesday, August 10, 2011
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

TDD
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need some of it to avoid re-inventing wheels.

TDD

ATDD
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

TDD

ATDD Daily standups
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

TDD

ATDD  Daily standups

Iterations
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

TDD

ATDD

Daily standups

Continuous integration

Iterations

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

---

TDD

ATDD

Daily standups

Retrospectives

Iterations

Continuous integration

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Is feed-forward waste?

Feed-forward using the Controller is application of *a priori* knowledge.
We always need *some* of it to avoid re-inventing wheels.

---

Diagram:

- Desired
- Error
- System input
- Actual
- Observed result
- Controller
- System
- Sensor

---

TDD

ATDD

Daily standups

Iterations

Continuous integration

Retrospectives

Deployment environments

---

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Noise

Signal
Signal + Noise

Wednesday, August 10, 2011
Noise

Signal + Noise = Huh?
Signals must be filtered from noise.

⇒ Delays

⇒ Lower communication bandwidth
Signals must be filtered from noise.
⇒ Delays
⇒ Lower communication bandwidth

In SW-development, noise can be
- Irrelevant information
- Disinformation
Exercise on feed-back system with delay

Time (s) 0 20 40 60 80 100

Desired

Measured

Robot

Controller

Tracker

Drop-zone

Wednesday, August 10, 2011
Exercise on feed-back system with delay

As before, but now three steps at a time!

Desired
Measured

Tracker
Robot
Drop-zone
Controller

Wednesday, August 10, 2011
Exercise on feed-back system with delay

As before, but now now *three steps at a time*!
Exercise on feed-back system with delay

As before, but now three steps at a time!

Desired
Measured

Time (s)

Controller
Robot
Drop-zone
Tracker

Wednesday, August 10, 2011

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com
Exercise on feed-back system with delay

As before, but now now *three steps at a time*!
Exercise on feed-back system with delay

As before, but now now three steps at a time!
Exercise on feed-back system with delay

As before, but now

now *three steps at a time*!
Exercise on feed-back system with delay

As before, but now now *three steps at a time*!
Exercise on feed-back system with delay

As before, but now now *three steps at a time!*

As before, but now now *three steps at a time!*

Exercise on feed-back system with delay

As before, but now now *three steps at a time!*
Exercise on feed-back system with delay

As before, but now now *three steps at a time!*

![Graph showing desired and measured values over time.]

- Desired
- Measured

Time (s)

---

Wednesday, August 10, 2011
Exercise on feed-back system with delay

As before, but now three steps at a time!
Exercise on feed-back system with delay

As before, but now
now *three steps at a time!*

![Diagram of system components: Desired and Measured values plotted over time, with symbols for Robot, Controller, Tracker, and Drop-zone.]
Exercise on feed-back system with delay

As before, but now three steps at a time!

Desired
Measured

Time (s)

Controller
Robot
Drop-zone
Tracker

Daniel Brolund — @danielbrolund — daniel.brolund@agical.com

Wednesday, August 10, 2011
Exercise on feed-back system with delay

As before, but now now *three steps at a time!*

Place sequence on back of Robot (FIFO queue). Give the first step to the robot, and add one step last in the queue.
Points of Discussion - groups of 4-6
Points of Discussion - groups of 4-6

Results in exercises?
Errors?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?

Errors?

What are the measurements we make?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?

What are the measurements we make?

Impact of robot precision?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?

What are the measurements we make?

Impact of robot precision?

What are the unknowns?
Points of Discussion - groups of 4-6

Results in exercises?  Difference between exercises?
Errors?

What are the measurements we make?

Impact of robot precision?

What are the unknowns?

Sensitivity to unknowns & changes? Robustness?
Points of Discussion - groups of 4-6

Results in exercises? Errors?

Difference between exercises?

What are the measurements we make?

What are the unknowns?

Impact of robot precision?

Iteration length?

Sensitivity to unknowns & changes? Robustness?
Points of Discussion - groups of 4-6

Results in exercises? Changes?
Errors?

What are the measurements we make?

Impact of robot precision?

Sensitivity to unknowns & changes? Robustness?

Difference between exercises?

What are the delays?

What are the unknowns?

Iteration length?
**Points of Discussion - groups of 4-6**

- Results in exercises? Errors?
- What are the measurements we make?
- Impact of robot precision?
- Sensitivity to unknowns & changes? Robustness?
- Difference between exercises?
- What are the delays?
- What are the unknowns?
- Iteration length?
- Impact of iteration length?
Points of Discussion - groups of 4-6

Results in exercises? Errors?

What are the measurements we make?

Impact of robot precision?

Sensitivity to unknowns & changes? Robustness?

Difference between exercises?

What are the delays?

What are the unknowns?

Iteration length?

Impact of iteration length?

Similarities with software development?
Exercise in group: Draw your feedback loops

[Diagram showing feedback loops]

TDD 10s
Retros 1w
Release 1m
"Softer" issues - discussion in groups

- What does the following do to your feedback loops?
  - Trust?
  - Distrust?
  - Lies?
  - Care?

- What is noise according to you?

- What other "soft" issues will affect the system? How?
Failure modes - discussion in groups

• How can you break the empirical process? What part?
  • Controller?
  • System?
  • Sensor?
  • Desired?
Conclusions & advice

• Introduce feedback where appropriate!
• Apply reasonable amount of a priori (feed-forward) knowledge.
• Tune the Sensors to improve resolution.
• Remove unnecessary Sensoring.
• Find the iteration cycle-time sweet-spot.
• Reduce delays to improve responsiveness, extend margins, or cheapen the Controller.
• Reduce noise.