An operational guide to monitorability

Luca Aceto
ICE-TCS, Department of Computer Science, Reykjavik University, and Gran Sasso Science Institute, L'Aquila

Tehran Institute for Advanced Studies
Cyberspace, 24 February 2021

General take-home message

Thou shalt

1. define notions of monitorability in terms of monitors,
2. view notions of monitorability as a spectrum, and
3. understand monitor guarantees.

General take-home message (in Icelandic runes)

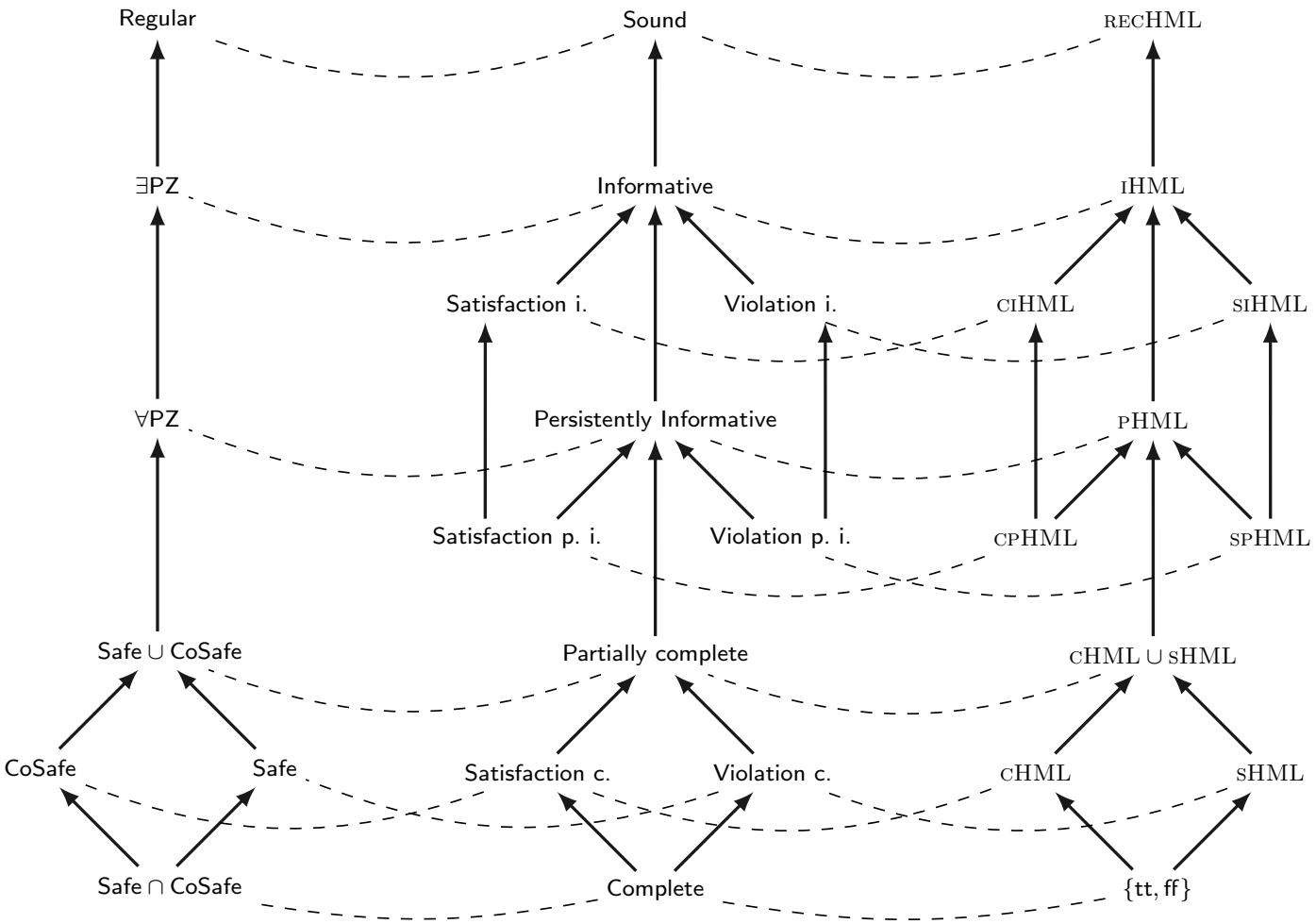
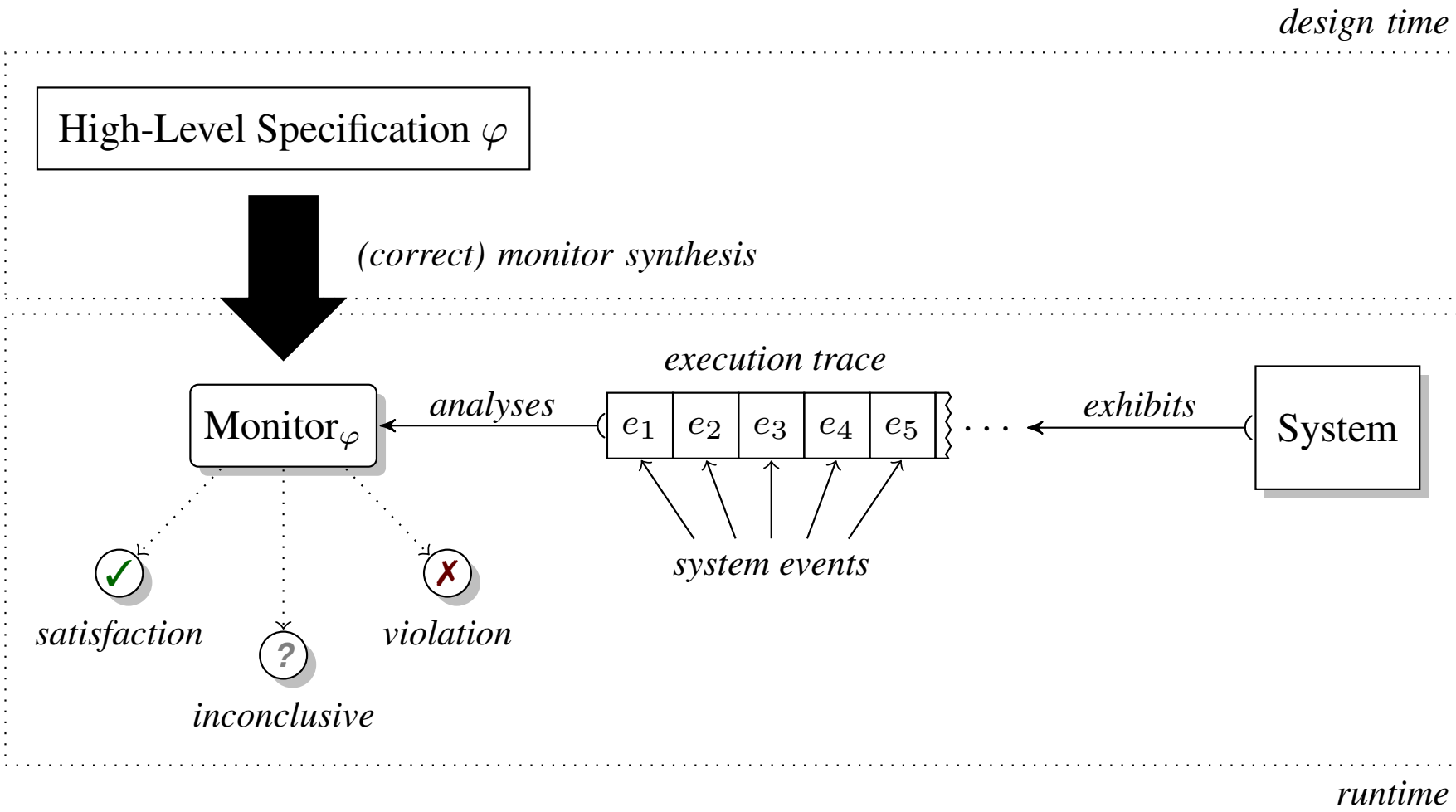


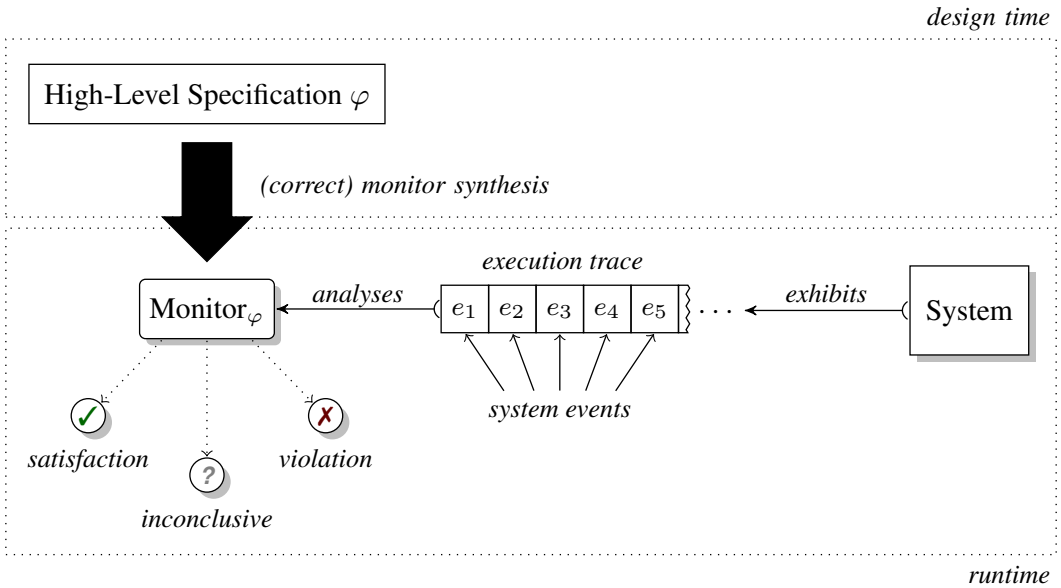
Fig. 1.1 The Monitorability Hierarchy of Regular Properties

Runtime monitoring in a nutshell



Assumption: Verdicts are irrevocable.

Runtime monitoring in a nutshell

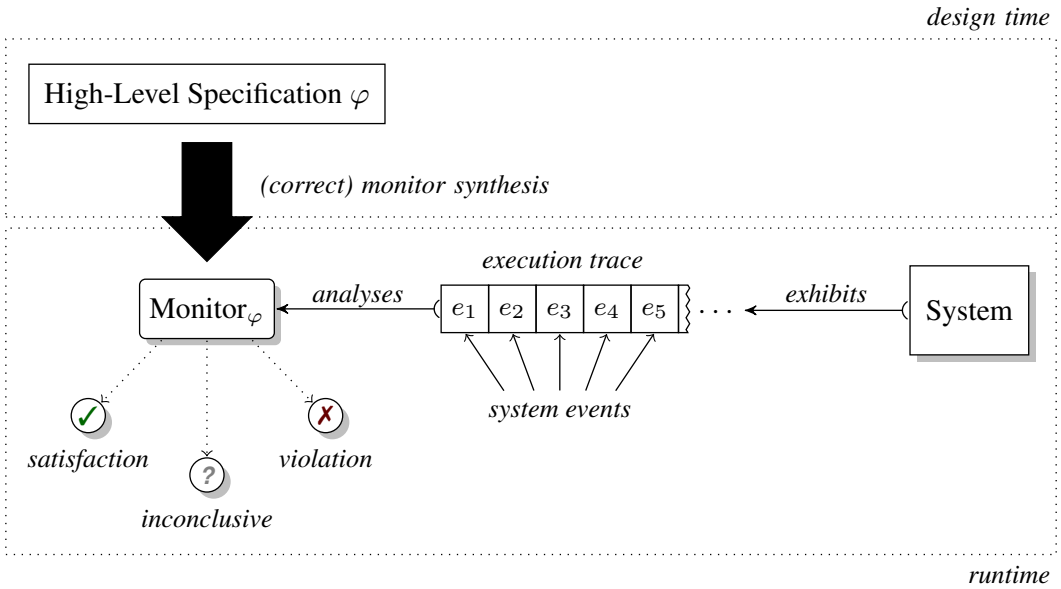


Why runtime monitoring?

Runtime monitoring is

- is lightweight and best effort,
- is post-deployment,
- can take advantage of hardware parallelism,
- can be applied to systems with ML components, cloud connectivity. . . .

Runtime monitoring in a nutshell



Key questions

1. When is a property monitorable? Characterizations?
2. What are monitors and what correctness guarantees do they give?
3. Can one synthesize 'correct' monitors from properties?

A yardstick notion: Pnueli-Zaks monitorability (2006)

Setting: Properties of finite and infinite traces over a finite set $\textsc{Act}$ of actions. We let $\textsc{Trc} = \textsc{Act}^* \cup \textsc{Act}^\omega$.

Definition

A property $P \subseteq \textsc{Trc}$ is *s*-monitorable, with $s \in \textsc{Act}^*$, if there is some $t \in \textsc{Act}^*$ such that P is 'positively or negatively determined by st'.

Example

The property

- ☹ *now and eventually* ☺, *or*
- *eventually always* ☺

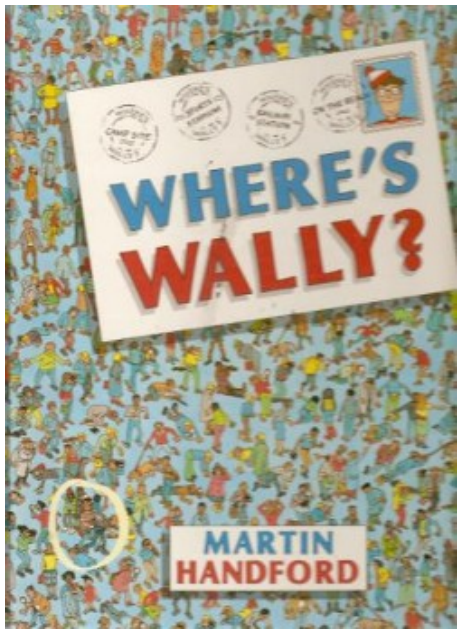
is *s*-monitorable for all strings that start with ☹ and for the empty string, but not for the others.

A yardstick notion: Pnueli-Zaks monitorability (2006)

Setting: Properties of finite and infinite traces over a finite set $\textsc{Act}$ of actions. We let $\textsc{Trc} = \textsc{Act}^* \cup \textsc{Act}^\omega$.

Definition

A property $P \subseteq \textsc{Trc}$ is *s*-monitorable, with $s \in \textsc{Act}^*$, if there is some $t \in \textsc{Act}^*$ such that P is 'positively or negatively determined by st'.



Our question

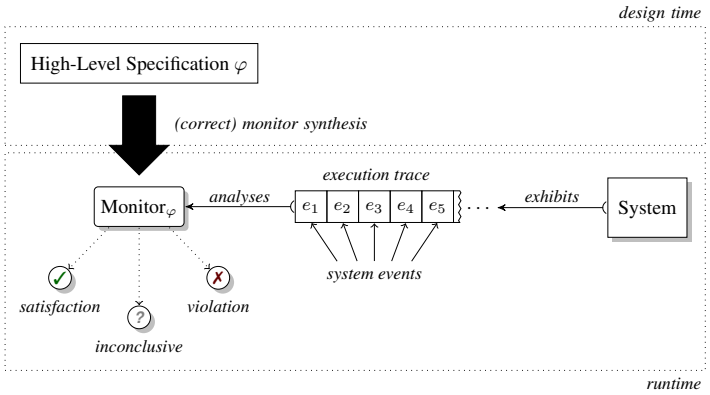
Where are the monitors?

Our motto: Bring back the monitors!

Take-home message (reloaded)

- Monitorability comes in a spectrum!
- Taking an operational view of monitorability, which defines it in terms of monitors and their correctness guarantees, allows us to develop a systematic theory of monitorability.

An operational guide to monitorability: Ingredients



Ingredients

- A formal specification language (our work: a fixed-point logic).
- A model of system behaviour (our work: (finite and) infinite traces, states in LTSs).
- A formalism for writing monitors (our work: (extensions of the) regular fragment of a CCS-like language/finite automata).
- Operational notions of instrumentation and monitorability.

Monitorability (operationally)

For a monitor m and a trace t,

$$\begin{aligned} \mathbf{acc}(m,t) &\stackrel{\text{def}}{=} m \text{ reports } \☺ \text{ when processing } t \\ \mathbf{rej}(m,t) &\stackrel{\text{def}}{=} m \text{ reports } \☹ \text{ when processing } t \end{aligned}$$

Logic

$t \in P$

$t \notin P$

correspondence? $\longleftrightarrow$

Monitoring

$\mathbf{acc}(m,t)$

$\mathbf{rej}(m,t)$

Monitorability (operationally)

For a monitor m and a trace t,

$$\begin{aligned} \mathbf{acc}(m,t) &\stackrel{\text{def}}{=} m \text{ reports } \☺ \text{ when processing } t \\ \mathbf{rej}(m,t) &\stackrel{\text{def}}{=} m \text{ reports } \☹ \text{ when processing } t \end{aligned}$$

Logic

$t \in P$

$t \notin P$

correspondence? $\longleftrightarrow$

Monitoring

$\mathbf{acc}(m,t)$

$\mathbf{rej}(m,t)$

Correctness guarantees: The ideal setting

- Soundness: Monitor m soundly monitors for P if 'its verdicts can always be trusted.'
- Completeness: Monitor m is complete for P if 'it provides all the valid verdicts.'

Towards a monitorability hierarchy: Levels of completeness

- Sound – everything has a sound monitor: 'I don't know'.
- ...
- ...
- ...
- Sound and Complete – only trivial properties have a sound and complete monitor: ☺ for True, ☹ for False.

A bit more than soundness

Definition (Informative monitors)

A monitor is informative if for some t, either $\mathbf{acc}(m, t)$ or $\mathbf{rej}(m, t)$.

A bit more than soundness

Definition (Informative monitors)

A monitor is informative if for some t, either $\mathbf{acc}(m, t)$ or $\mathbf{rej}(m, t)$.

With and without informative monitors

- $b.\☹$ is sound and informative for 'always and forever a'.

A bit more than soundness

Definition (Informative monitors)

A monitor is informative if for some t, either $\mathbf{acc}(m, t)$ or $\mathbf{rej}(m, t)$.

With and without informative monitors

- $b.\☹$ is sound and informative for 'always and forever a'.
- The property 'eventually always b' has no sound and informative monitor.

A bit more than soundness

Definition (Informative monitors)

A monitor is informative if for some t, either $\mathbf{acc}(m, t)$ or $\mathbf{rej}(m, t)$.

With and without informative monitors

- $b.\☹$ is sound and informative for 'always and forever a'.
- The property 'eventually always b' has no sound and informative monitor.

Definition (Informative monitorability)

A property is **informatively monitorable** if it has a sound and informative monitor.

Levels of completeness (take 2)

- Sound – everything has a sound monitor: 'I don't know'.
- Informative
- ...
- ...
- Sound and Complete – only trivial properties have a sound and complete monitor: ☺ for True, ☹ for False.

A weaker completeness

Definition (Violation completeness)

Monitor m is a **violation-complete** monitor for the property P, if for all traces $t \in \text{Act}^* \cup \text{Act}^\omega$ we have:

- $t \notin P$ implies $\mathbf{rej}(m, t)$.

A weaker completeness

Definition (Violation completeness)

Monitor m is a **violation-complete** monitor for the property P, if for all traces $t \in \text{Act}^* \cup \text{Act}^\omega$ we have:

- $t \notin P$ implies $\mathbf{rej}(m, t)$.

Definition (Violation monitorability)

P is violation monitorable if it has a sound and violation-complete monitor.

Satisfaction-complete monitors and satisfaction monitorability are defined in the natural way.

Completeness at work: Examples

Violation or satisfaction complete?

1. $a.\☹$ **sound and violation complete for** 'doesn't start with a'.
2. $a.\☺$ **sound and satisfaction complete for** 'starts with a'.

Completeness at work: Examples

Violation or satisfaction complete?

1. $a.\☹$ **sound and violation complete for** 'doesn't start with a'.
2. $a.\☺$ **sound and satisfaction complete for** 'starts with a'.
3. $a.\☹$ **not** violation complete for 'starts neither with a nor with b'.

Completeness at work: Examples

Violation or satisfaction complete?

1. $a.\☹$ **sound and violation complete for** 'doesn't start with a'.
2. $a.\☺$ **sound and satisfaction complete for** 'starts with a'.
3. $a.\☹$ **not** violation complete for 'starts neither with a nor with b'.
4. $a.\☹ + b.\☹$ **sound and violation complete for** 'starts neither with a nor with b'.

Levels of completeness (take 3)

- Sound – everything has a sound monitor: 'I don't know'.
- Informative – Existential Pnueli-Zaks
- Persistently informative – Universal Pnueli-Zaks
- Sound and either violation- or satisfaction-complete – Safety and co-safety properties
- Sound and Complete – only trivial properties have a sound and complete monitor: ☺ for True, ☹ for False.

Levels of completeness (take 3)

- Sound – everything has a sound monitor: 'I don't know'.
- Informative – Existential Pnueli-Zaks
- Persistently informative – Universal Pnueli-Zaks
- Sound and either violation- or satisfaction-complete – Safety and co-safety properties
- Sound and Complete – only trivial properties have a sound and complete monitor: ☺ for True, ☹ for False.

Addendum 1: The joys of syntactic characterizations (for regular properties)

Example: Safety informative property $= \varphi_1 \wedge \varphi_2$, where φ_1 is in the 'safety fragment and contains false'. See

`http://icetcs.ru.is/theofomon/SoSym.pdf`.

Levels of completeness (take 3)

- Sound – everything has a sound monitor: 'I don't know'.
- Informative – Existential Pnueli-Zaks
- Persistently informative – Universal Pnueli-Zaks
- Sound and either violation- or satisfaction-complete – Safety and co-safety properties
- Sound and Complete – only trivial properties have a sound and complete monitor: ☺ for True, ☹ for False.

Addendum 2: Monitorability depends on the semantic domain

Over infinite traces, all modal properties have sound and complete monitors!

Further results

- Correct-by-design, monitor-synthesis functions for 'monitorable properties' expressed in our touchstone logic.
- Branching-time monitorability and its relations to linear-time one.
- Power of deterministic and parallel monitors: The cost of monitoring deterministically and/or alone.
- Monitoring the unmonitorable.
- Tool detectEr for monitoring Erlang programs.
- Runtime enforcement.

Projects TheoFoMon (2016–2020) and MoVeMnt (2021–2023)

Follow `http://icetcs.ru.is/theofomon/` and `https://sites.google.com/view/antonisachilleos/movemnt`!

Some future research directions

Apply our methodology to

- distributed runtime monitoring/enforcement,
- logics over multiple traces,
- probabilistic/real-time/cyber-physical/smart systems,
- monitoring and 'learning' ...

Study the relationships between logics of knowledge and monitoring.

Big Brothers and Sisters at Reykjavik University (and elsewhere)








Take-home message (reloaded)

- Monitorability comes in a spectrum!
- Taking an operational view allows us to develop a systematic theory of monitorability and monitor correctness.

Thank you!