Technische Universität Berlin

Programmer Memory Failure and Interruptions

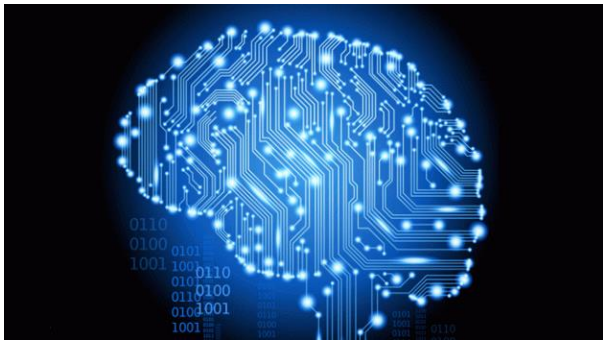
Presenter: Christopher O‘Hara | Supervisor : Stefan Sydow

Software Engineering of Embedded Systems | 21.1.2019

Motivation:

Why create memory aids?

- Memory directly impacts performance
- Humans are highly susceptible to interruptions
- Memory faults lead to an increase in errors
  - Approximately twice the errors
- Increases time to complete tasks
  - Approximately twice as long



geek.com [1]

Current frameworks are out of date.

Technische Universität Berlin

Research:

[1] Parnin et al., *"Programmer Information Needs After Memory Failure"*, IEEE (2012)

- Discusses causes for memory failure and how to address them for task completion

[2] Zahra Shakeri Hossein Abad et al., "A Visual Narrative Path from Switching to Resuming a Requirements Engineering Task" IEEE (2017)

- Discusses causes for interruptions and how to address them for task completion

SESE

Technische Universität Berlin

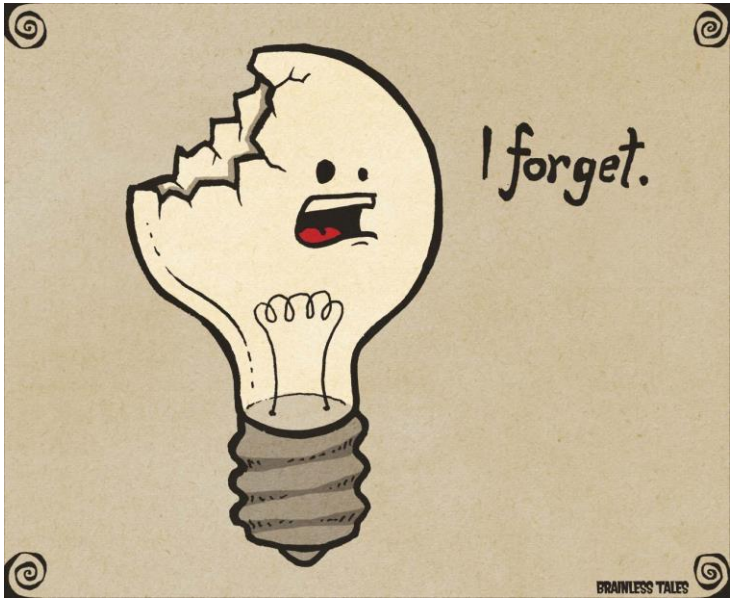
Programmer Information Needs After Memory Failure :

Goals / Approach:

- Explore differences in human memory abilities
- Categorizes programmer tasks
- Proposes *Worklets* for implementation
- Lay the foundation for further work

Results:

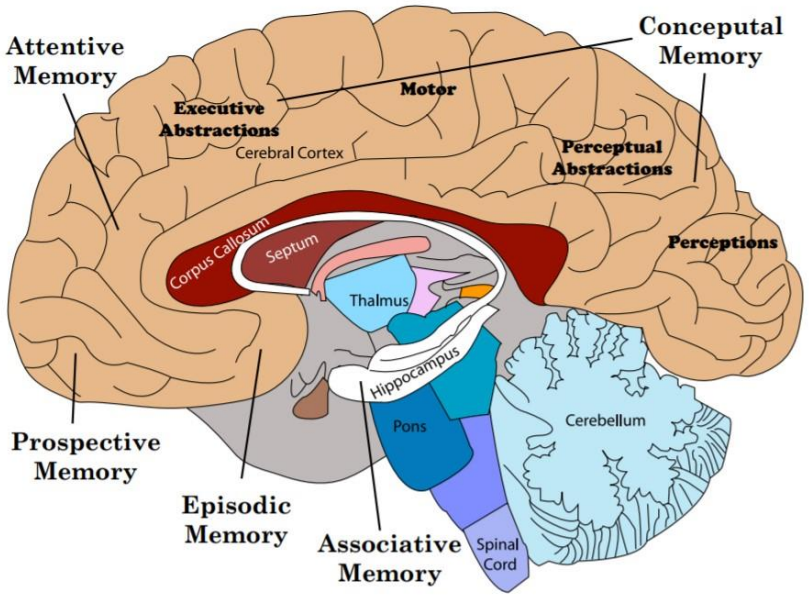
- Five “memory aids” targeting memory faults



brainlesstales.com [2]

SESE

Memory and the Brain

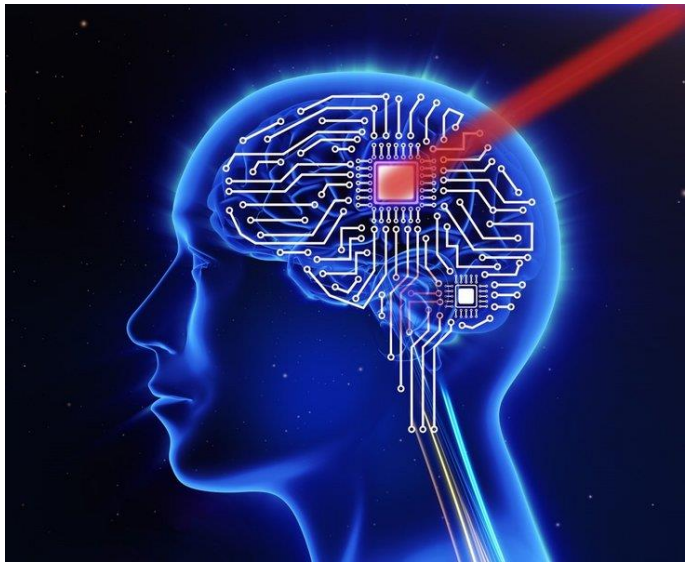


Parnin et al. [1]

- Five primary types of memory:
  - Attentive
  - Prospective
  - Episodic
  - Associative
  - Conceptual
- Memory triggered by different components determines shelf-life
- Complementary functions
- Effectiveness components of intent, interest, and impact of event / stimuli

Technische Universität Berlin

Intuitive Memories



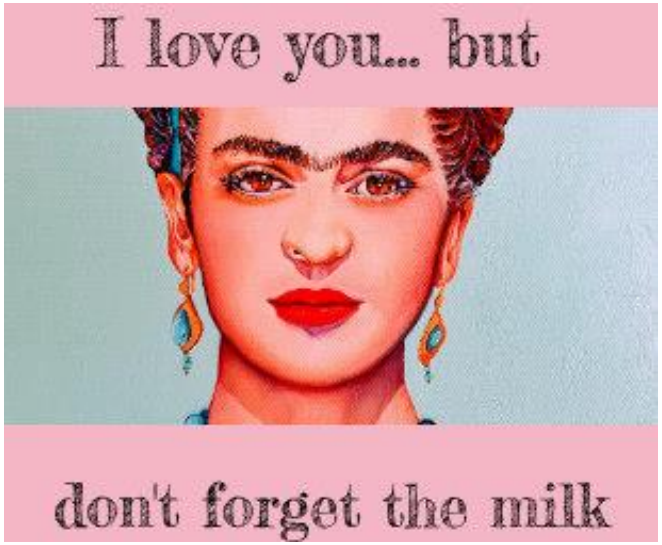
http://ingenioer.au.dk [3]

Human-Computer Analogy

- Types
  - Attentive
  - Prospective
  - Episodic
- Computer Analogy
  - RAM / Flash Memory
  - Hard drives
  - Event-driven or time-driven tasks
  - Organic sensor network

SESE

Memory: Prospective



rlv.zcache.com [4]

"Remember to buy milk on the way home."

- Usage:
  - Future actions
  - Specific circumstances
- Failures:
  - Monitor
  - Engage
- Operations:
  - Retrieving memory via environmental cues

Technische Universität Berlin

Abstract Memories

tutsplus.com [5]

"This reminds me of a [x]."

SESE

Abstract Memories

Types:

- Associative
- Conceptual

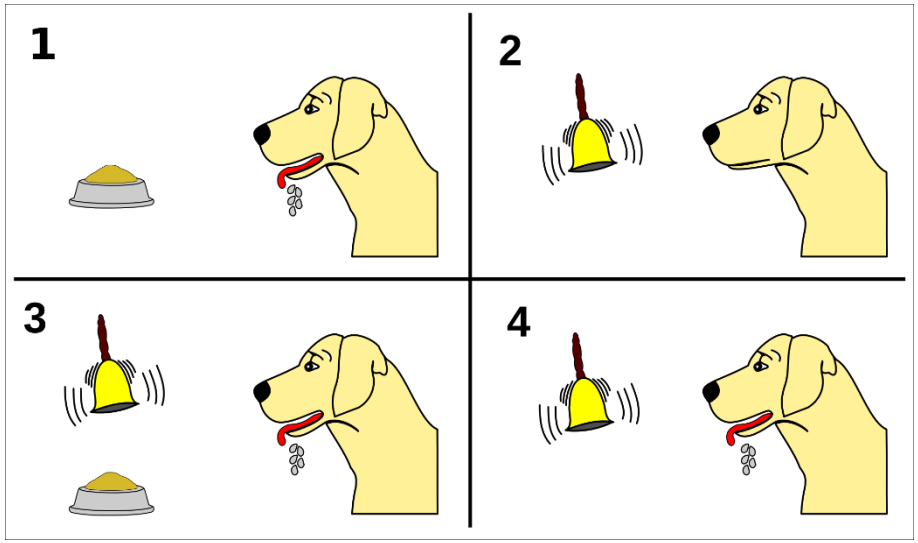
Computer Analogy:

- No current implementations

tutsplus.com [5]

"This reminds me of a ball."

Memory: Associative



wikipedia.org [6]

"I'll recognize it when I see it."

- Usage:
  - Automatic recording of events
  - Foundation for LTM
- Failures
  - Retention
  - Association
- Operations
  - Creating associations based on impact
  - Attaching to memory framework

Technische Universität Berlin

Programmer Needs

- Prospective:
  - “Programmers need facilities for modulating levels of engagement”
- Attentive:
  - “Developers need support for persistent focus”
- Associative:
  - “Developers need support for building associations in code locations”
- Episodic:
  - “Developers need support in recalling personal and social narratives”
- Conceptual:
  - “Programmers need support in relevant concepts to promote priming”

SESE

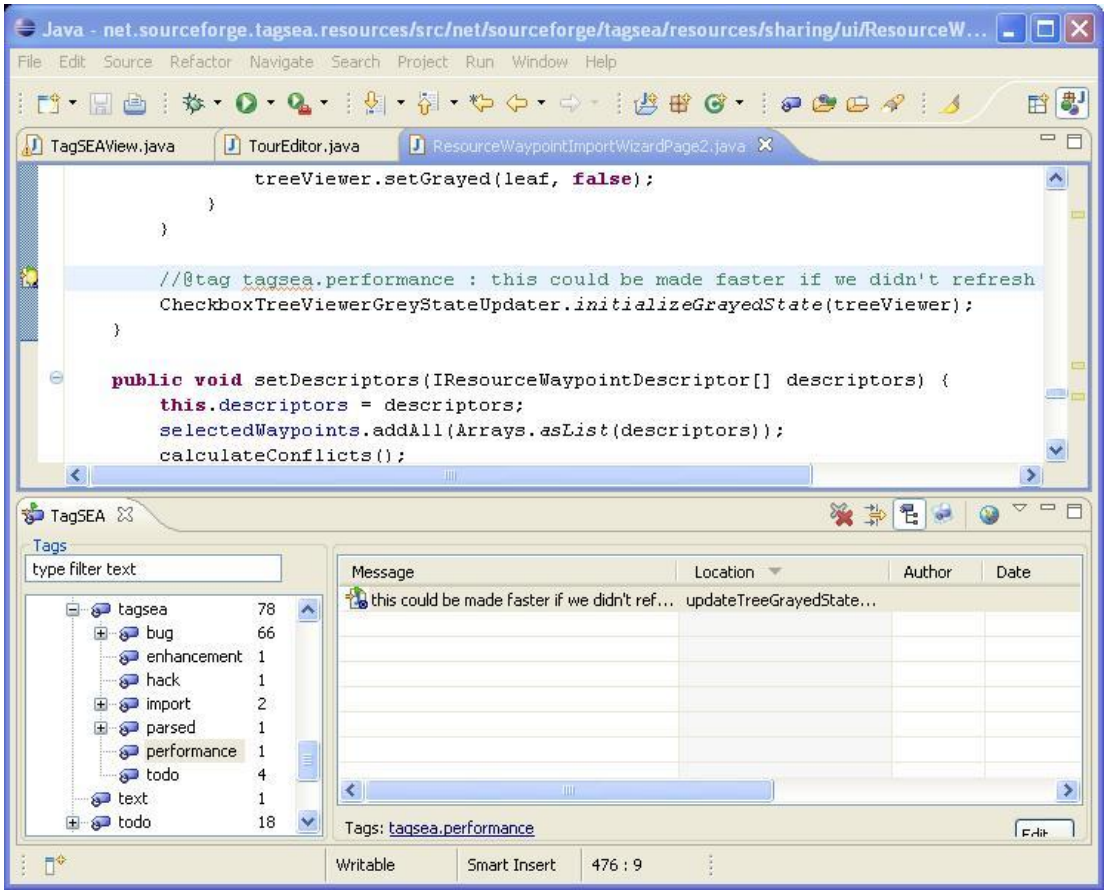
Information Needs and Memory Aids

www.pinterest.com [7]

- Prospective Activity:
  - Resuming blocked tasks
- Failures
  - Monitor failure
  - Engage failure
- Information Needs
  - Multi-level engagement
  - Support monitoring applicability
- Memory Aids
  - “Smart Reminders” (TagSea)

Technische Universität Berlin

TagSea



tagsea [8]

SESE

Technische Universität Berlin

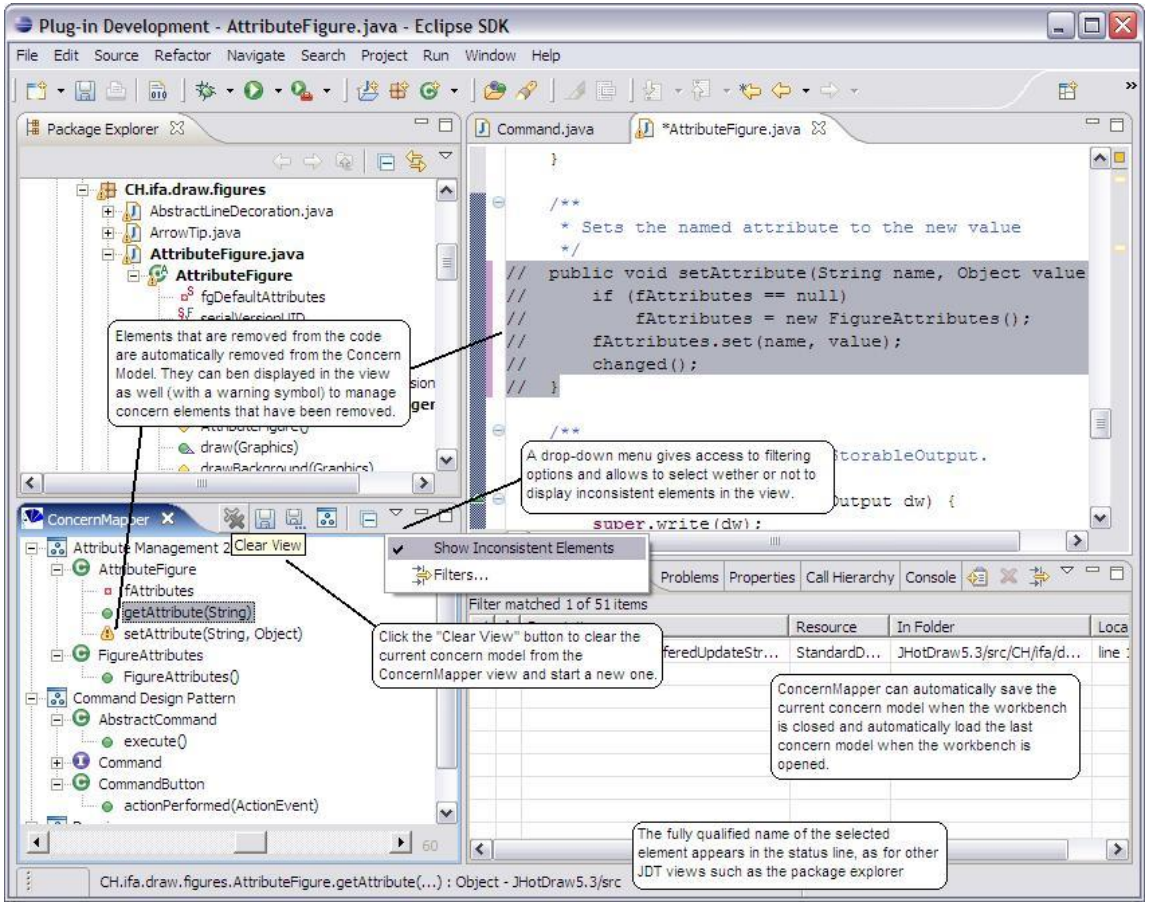
Information Needs and Memory Aids

lawnswood.org [9]

- Conceptual Activity:
  - Forming concepts
- Failures
  - Activation failure
  - Formation failure
- Information Needs
  - Support priming
  - Support abstraction
- Memory Aids
  - “ConcernMapper”

SESE

ConcernMapper



concernmapper [10]

Technische Universität Berlin

Conclusion: Programming Information Needs After Memory Failure

- Worklets are potentially useful to increase productivity for programmers
- Different types of memory require different strategies
- Authors primarily rely on integration to Visual Studio / Eclipse

Question for the future: Can we use something else?

SESE

Technische Universität Berlin

A Visual Narrative Path from Switching to Resuming a Requirements Engineering Task

Goals / Approach:

- Determine causes of interruptions
- Categorize interruption process
- Proposes *Interruption Visualization Framework*
- Integrate work from Parnin et al. and others

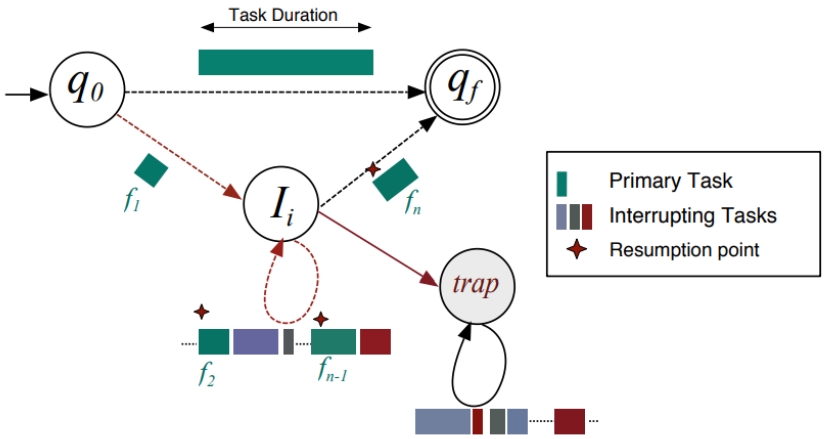
Results:

- Programmers require time to remember
- 11% of tasks are never resumed
- Different visualizations = different results
- Reminders are not useful if they arrive at inappropriate times

monash.edu [11]

SESE

Task Interruptions



Abad et al. [2]

- Three key topics of interruptions:
  - Disruptiveness
  - Interruption Characteristics
  - Self-Interruptions
- Three timeliness events:
  - Before Interruption
  - Suspension Period
  - After Interruption
- Visualizations can be in the form of “data”, “information”, and “knowledge

Technische Universität Berlin

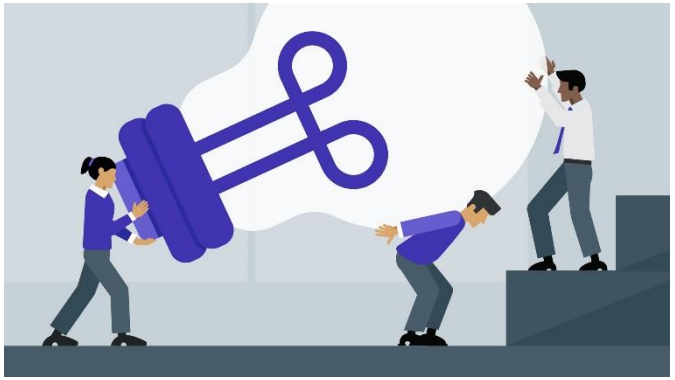
Interruption Causes

- Task-switching (30%)
  - Tasks get blocked via lack of information, waiting for feedback
- Re-prioritization (26%)
  - Other topics temporarily gain priority
- Self-Interruption (39%)
  - 22% is from “getting bored”
  - 17% is from “personal schedules”
- Other (5%)
  - “I usually get distracted by researching a new technology that could help solve the task at hand.”

SESE

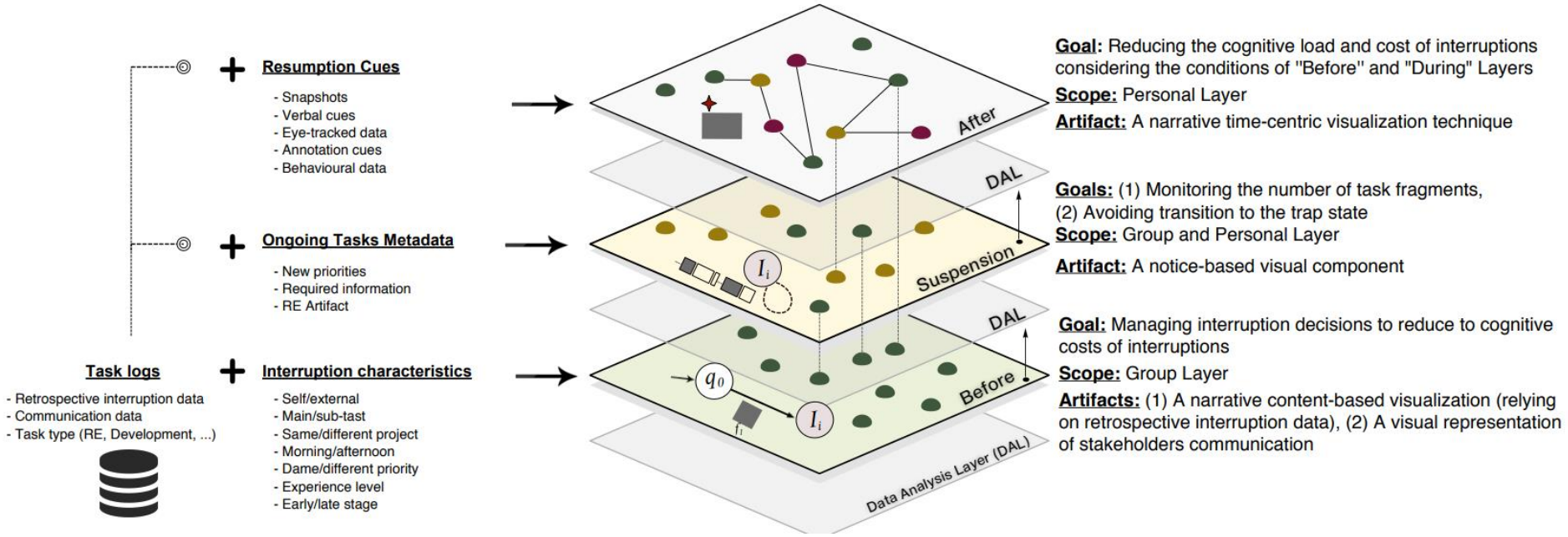
Layers

- Personal Layer
  - Goal: reduce cognitive load of interruptions
  - Artifact: A narrative time-centric visualization
- Group & Personal Layer
  - Goal: Monitoring number of task fragments
  - Goal: Avoiding transition into “trap state”
  - Artifact: A notice-based component
- Group Layer
  - Goal: Managing interruption decisions
  - Artifact: A narrative content-based visualization
  - Artifact: A visual representation of stakeholder communication



lynda.com [11]

Layers



Abad et al. [2]

Interruption Aids

- Notifications (81%)
  - Pop-ups, verbal notifications, sound effects
- Visual (19%)
  - Image of the interrupted task, open tabs on IDE

"It should appear when I'm between tasks, or at least at a reasonable pausing point…"

wpmudev.org [13]

Technische Universität Berlin

Conclusion: A Visual Narrative Path from Switching to Resuming a Requirements Engineering Task

- Aids at an appropriate time would be the most effective
- Breaking issues into layers can assist in approaching interruptions
- Self-interruptions are a serious problem

Question for the future: Can we use something else?

SESE

Technische Universität Berlin

Worklets vs *Layers*: Comparison

Worklets

- Integrate into IDE
- Improve task resumption
- Memory-focused

In implementation:

- No follow-up data

Layers

- Augment more than programming
- Improve task resumption
- Cause-effect focused

In implementation:

- Currently assessing effectiveness on resumption time

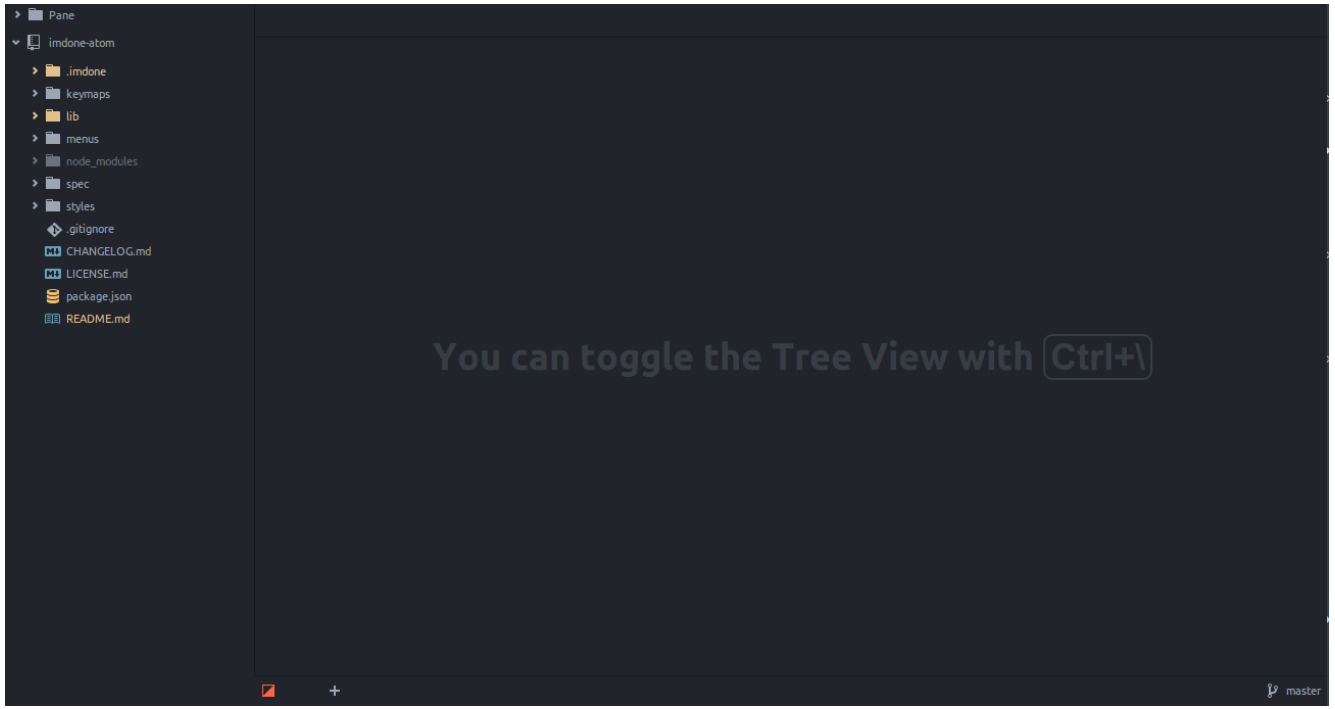
SESE

Other Strategies

```
33  31
34  32    void FusionEkf::ProcessMeasurement(SensorReading& reading) {
35  33      if (!is_initialized_) {
36      -     //setup efk P
37      -     ekf_.P_ = MatrixXd(4, 4);
38      -     ekf_.P_ << 1, 0, 0, 0,
39      -                0, 1, 0, 0,
40      -                0, 0, 1000, 0,
41      -                0, 0, 0, 1000;
42      -
43      -     ekf_.x_ = VectorXd(4);
44      -
45  34        //we will need to init the state vector x,y,vx,vy
46  35        switch (reading.sensor_type) {
47  36          case SensorType::LASER:
    37  +         ekf_.x_ = VectorXd(4);
    38  +         ekf_.x_ << reading.measurement[0], reading.measurement[1], 0, 0;
48  39            break;
49  40          case SensorType::RADAR:
    41  +         ekf_.x_ = utility::PolarToCartesian(reading.measurement);
50  42            break;
51  43        }
```

github [14]

//TODO


Pane
imdone-atom
.imdone
keymaps
lib
menus
node_modules
spec
styles
.gitignore
CHANGELOG.md
LICENSE.md
package.json
README.md
You can toggle the Tree View with Ctrl+\
master

atom [15]

Technische Universität Berlin

Auto-Complete

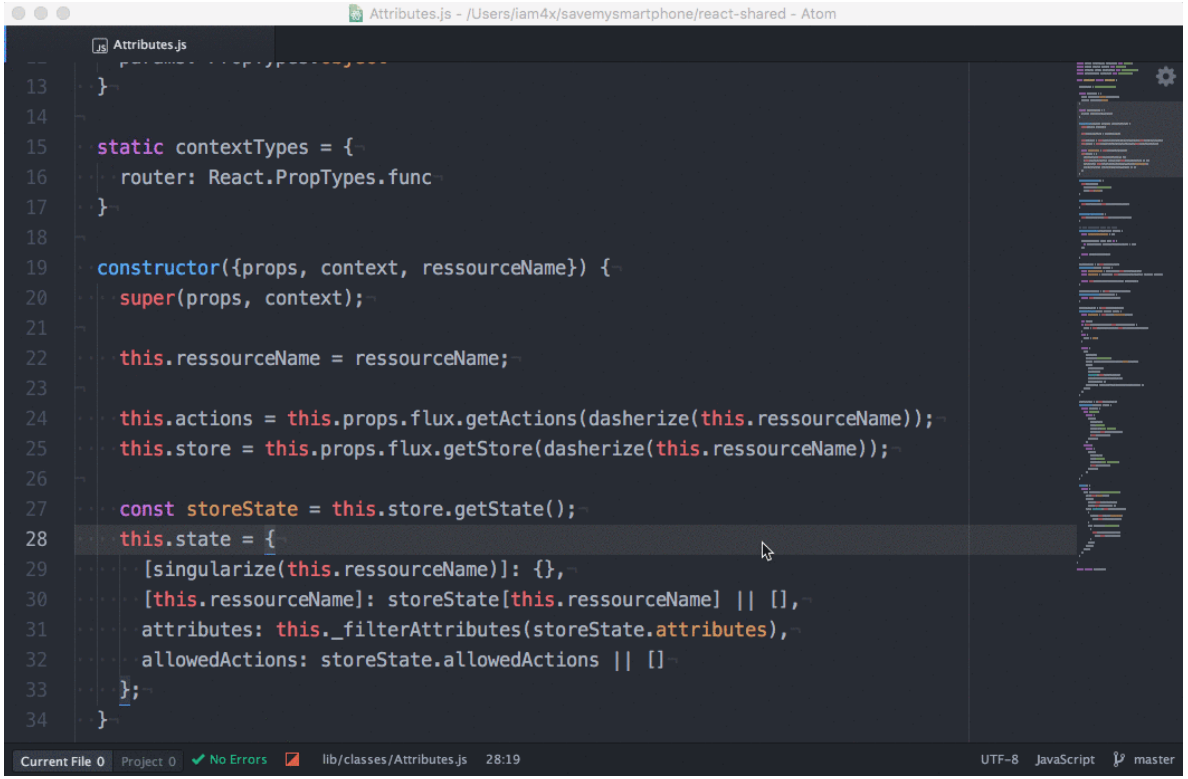
```
    local sceneGroup = self.view

    I

    -- Called when the scene's view does not exist
    --
    -- INSERT code here to initialize the scene
    -- e.g. add display objects to 'sceneGroup', add touch listeners, etc
end

function scene:show( event )
    local sceneGroup = self.view

    local phase = event.phase
```

atom [15]

SESE

Technische Universität Berlin

Error-Prevention



atom [15]

SESE

Technische Universität Berlin

Error-Prevention

```
112     this.controller.login('member', {
113       email: 'foo@bar.com'
114       password: 'foobarpassword',
115       remember: false
116     });
117     this.server.respond([200, {}, '']);
118     this.spy.should.have.been.calledWith('auth:login:ok');
119   });
120
121   it('should send an event nok on login error', function () {
122     var spy = sinon.spy(vent, 'trigger');
```

jshint Expected 'password' to have an indentation at 7 instead at 9.

jshint Expected '}' to match '{' from line 112 and instead saw 'password'.

jshint Expected ')' and instead saw ':'.

jshint Missing semicolon.

jshint Expected 'foobarpassword' to have an indentation at 7 instead at 19.

test/spec/modules/auth/controllers/api.spec.js* 114,29 JavaScript

atom [15]

SESE

Technische Universität Berlin

Conclusion: Assessment / Future Work

- Ideal future research will analyze the effectiveness of aids
- It is possible to mitigate memory failure and interruption costs
- Potential to drastically improve efficiency
- Also valid for non-programming tasks

SESE

Technische Universität Berlin

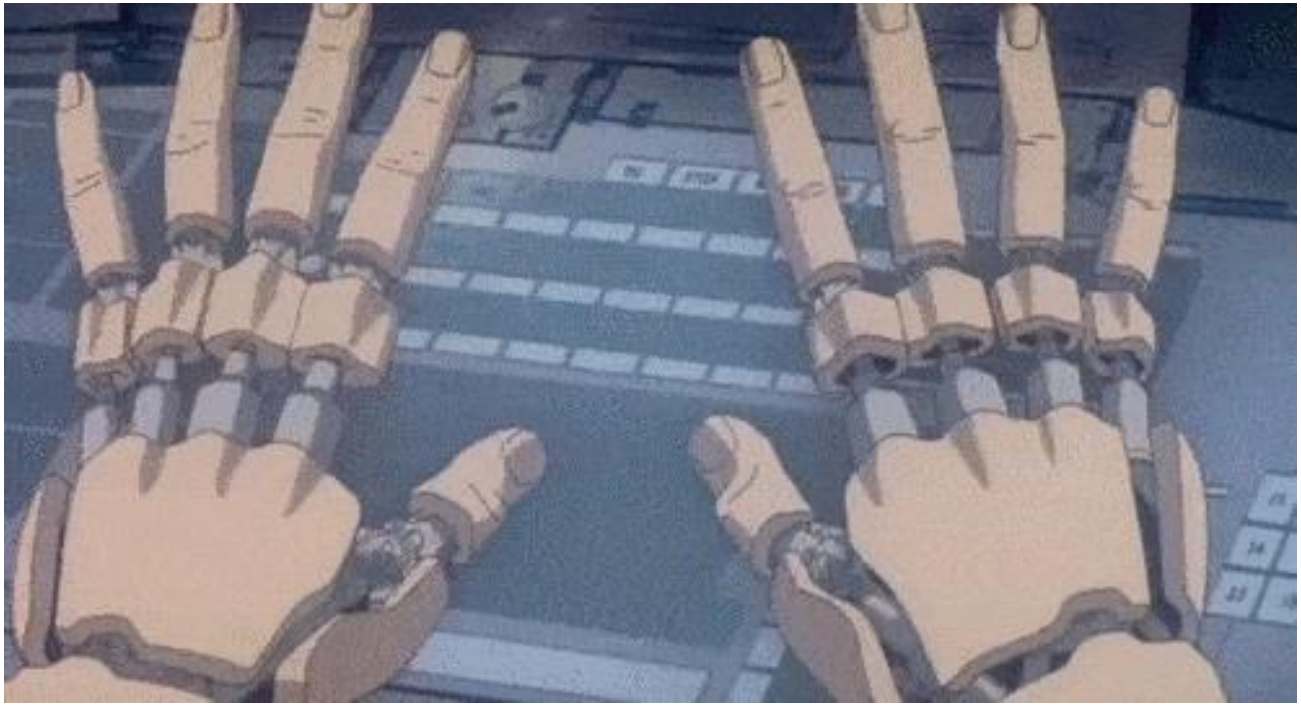
Present:

gifer.com [16]

SESE

Technische Universität Berlin

Goal:



medium [15]

Technische Universität Berlin

Thank you for your attention.

Questions?

SESE

Technische Universität Berlin

References:

[1] Parnin et al., *"Programmer Information Needs After Memory Failure"*, IEEE 20th IEEE International Conference of Program Comprehension (ICPC) (2012)
IEEE/ISCA (2016)

[2] Zahra Shakeri Hossein Abad et al., *"A Visual Narrative Path from Switching to Resuming a Requirements Engineering Task"*, IEEE/RE 25th International Requirements Engineering Conference (2017)

SESE

Technische Universität Berlin

Image References:

[1] https://www.geek.com/new/googles-deepmind-creats-a-computer-that-mimics-human-short-term-memory-1608244
[2] www.brainlesstales.com/2013-10-22/i-forget
[3] http://ingenioer.au.dk/en/current/news/view/artikel/building-a-computer-like-a-human-brain-a-technological-revolution/
[4] https://rlv.zcache.com/i_love_you_but_dont_forget_the_milk_pink_magnet-r4368bc27d7284f4db7f32f4e0ba5aaae_x7j3u_8byvr_307.jpg?rvtype=content
[5] https://design.tutsplus.com/tutorials/animation-for-beginners-how-to-animate-a-bouncing-ball--cms-24787
[6]https://upload.wikimedia.org/wikipedia/commons/thumb/7/70/Pavlov%27s_dog_conditioning.svg/2000px-Pavlov%27s_dog_conditioning.svg.png
[7] https://www.pinterest.com/MarkO_The_App/marko-smart-reminder/?lp=true
[8] http://tagsea.sourceforge.net/images/tagsea.jpg
[9] http://www.lawnswood.org.uk/reminder-to-all-students/
[10] https://www.cs.mcgill.ca/~martin/cm/cm-1.2.0.JPG

SESE

Technische Universität Berlin

Image References:

[11] https://www.monash.edu.my/library/about/news/2015/articles/service-interruption
[12] https://www.lynda.com/Business-Skills-tutorials/Teamwork-Fundamentals/365728-2.html
[13] https://premium.wpmudev.org/project/the-pop-over-plugin/
[14] https://github.com/Ohara124c41
[15] https://github.com/mehcode/awesome-atom
[16] https://gifer.com/en/
[17] https://medium.com/buzzfeed-design/design-tooling-with-solid-atom-or-making-the-ide-our-bff-d2b0035cf62d

SESE

Technische Universität Berlin

TABLE I: INFORMATION NEEDS AND MEMORY AIDS FOR DIFFERENT MEMORY FAILURES.

MEMORY	PROGRAMMING ACTIVITY	FAILURE	INFORMATION NEED	MEMORY AIDS
prospective	Resuming blocked tasks	Monitor failure Engage failure	Support monitoring applicability Provide multi. levels of engagement	*smart reminders*
attentive	Refactoring large code	Concentration failure Limit failure	Provide persisted and stateful focus Facilitate multiplicity	*touch points*
associative	Navigating unfamiliar code	Retention failure Association failure	Provide distinguishable features Support indexing by multi. modalities	*associative links*
episodic	Learning new API	Source failure Recollection failure	Store context Support narrative	*code narratives*
conceptual	Forming concepts	Activation failure Formation failure	Support priming Support abstraction	*memlets*

SESE