Persistency Management in CEP

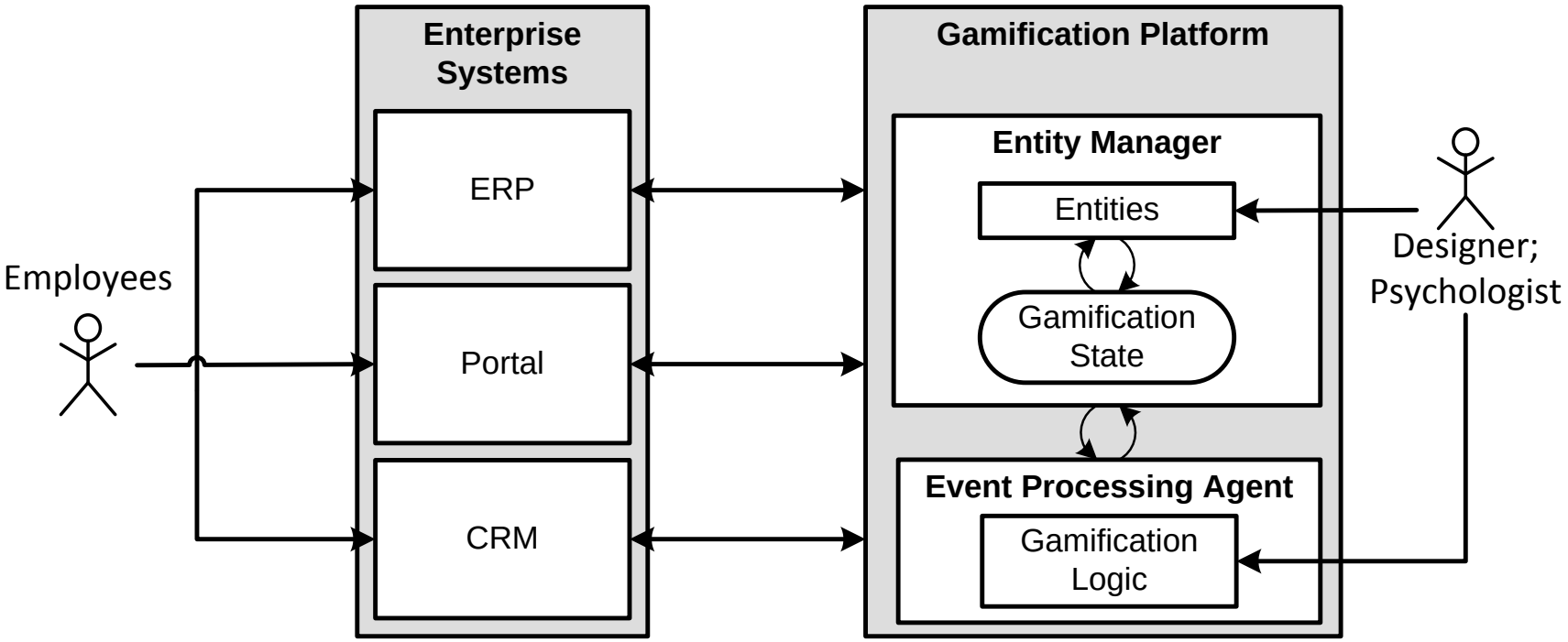
Hybrid Approach for Gamification Systems

Philipp Herzig, SAP AG, 09.07.2013

Introduction
Context in CEP [EN11]

- Temporal
  - Fixed Interval, Event Interval, Sliding fixed interval, sliding event interval
- Spatial
  - fixed, entity, event
- Segmentation-oriented
  - attribute list, partition identifier, stratification
- **State-oriented**
  - **entity-based [OSS+11]**
  - ontology-based [TRP12]

Application Scenario
Gamification System



Requirements

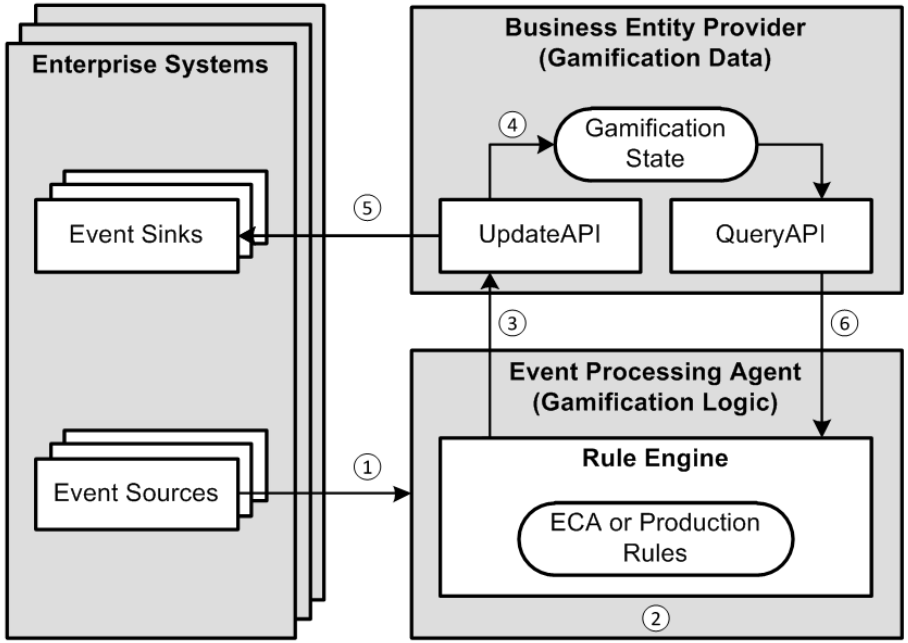
- Flexibility
  - Logic, Rule Language, Entity Behavior must be easy to change
- Real-time pattern detection
  - Detection of user situations at least in soft real-time
- Persistency
  - User progress has to be stored for later retrieval and update
- Analyzeability (e.g., ex-post or ad-hoc queries)
  - The success of gamification should be analyzeable across various dimensions such as time, users, or groups
- Manageability
  - Persisted data has to be managed over the entire lifecycle (e.g., anonymization, backup aggregation, composition)

Solution Approaches

Requirement	CEP/BRMS	Database	Hybrid System
Flexibility	**Yes**	No	**Yes**
Real-Time Event Correlation and Detection	**Yes**	No	**Yes**
Persistency	No	**Yes**	**Yes**
Analyzability	No	**Yes**	**Yes**
Manageability (e.g., Backup, Anonymization)	No	**Yes**	**Yes**

Gamification Platform

Hybrid Approach (based on [OSS+12])



Example	Formal Notation
User	U_1
Set of users	$\{U_1, ..., U_k\}$
Set of points for User U_i	$\{p_{i1}, ..., p_{in}\}$
Set of badges for User U_i	$\{b_{i1}, ..., b_{im}\}$
Sum of points for User U_i at timestamp t	$\mathcal{P}_{t,U_i} = \sum_{j=0}^{n} p_{ij}$
Average of points for User U_i	$\overline{\mathcal{P}_{U_i}} = \frac{1}{n} \sum_{j=0}^{n} p_{ij}$
Individual high score for user U_i based on $\mathcal{P}$	$(\mathcal{P}_{t,U_i}, ..., \mathcal{P}_{t+k,U_i})$
Leaderboard between users	$(U_1, ..., U_k)$

LHS Types	Example
(a) Simple Event / Event Rule	$e_1 \rightarrow ...$
(b) Boolean event correlation	$e_1 \wedge e_2 \rightarrow ...$
(c) Temporal event operators	$e_1\ during\ e_2 \rightarrow ...$
(d) Event Aggregation	$\frac{1}{n} \sum_{i=0}^{n} e_i^{value} > 20 \wedge e_2 \rightarrow ...,\ n = sizeof(window,$
(e) Event with Context	$e_1 \wedge \mathcal{P}_{t,U_i} \geq 20 \rightarrow ...$
(f) Context only	$U_i \wedge (b_{i1} \vee b_{i2}) \rightarrow ...$

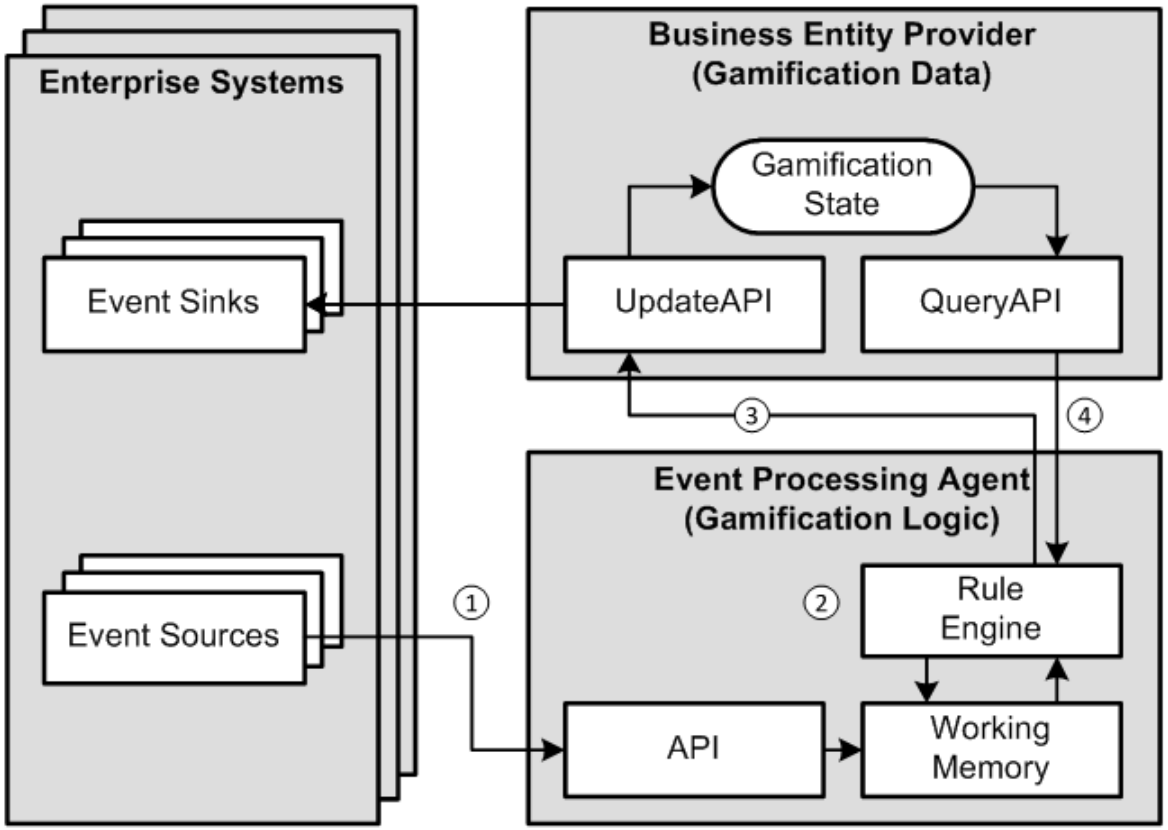
RHS Types	Example
(g) Multiple Events	$... \rightarrow e_2, e_3$
(h) Multiple Data (e.g., Point or Badge)	$... \rightarrow p_{i1}, b_{i2}$
(i) Multiple Data and Events	$... \rightarrow p_{i2}, b_{i1}, e_2, e_3$

Hybrid Approach
Example

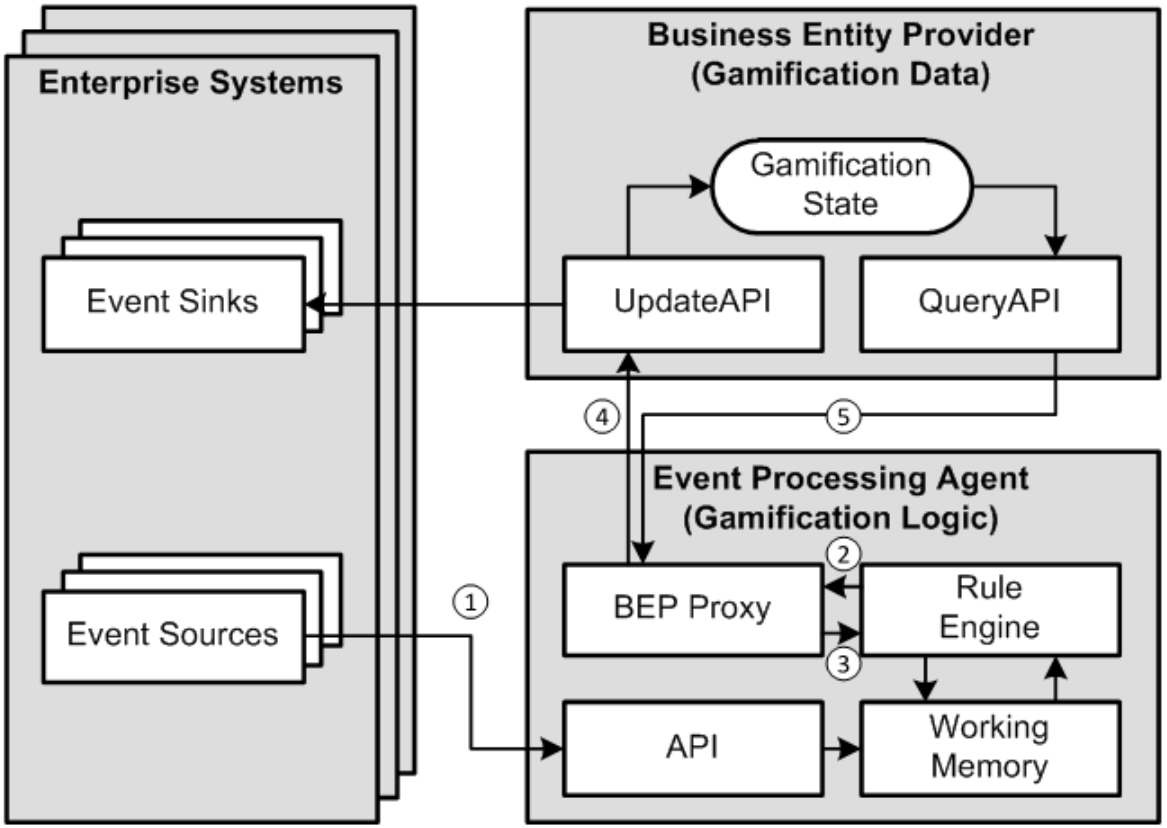
```
1 rule "AddedBuddyAndAddedTag"
2   when
3     $addBuddy : EventObject(type=='addBuddy', $playerid:playerid) from entry-
      point eventstream
4     $addTag : EventObject(type=='addTag', playerid==$playerid) from entry-point
      eventstream
5   then
6     updateAPI.givePoints($playerid, 'Experience', 1, 'TestReason');
7 end
```

```
10 rule "tenBuddies"
11   when
12     p : Player($playerid : uid)
13     eval(queryAPI.hasPlayerMission($playerid, 'I Have Got Buds!') == true)
14     eval(queryAPI.getPointsForPlayer($playerid, 'Buddies').getAmount() >= 10)
15   then
16     updateAPI.completeMission($playerid, 'I Have Got Buds!');
17     update($p); //only in synchronous mode
18 end
```

Hybrid Approach
Synchronous Communication



Hybrid Approach
Asynchronous Communication



Hybrid Approach
Experimental Setup

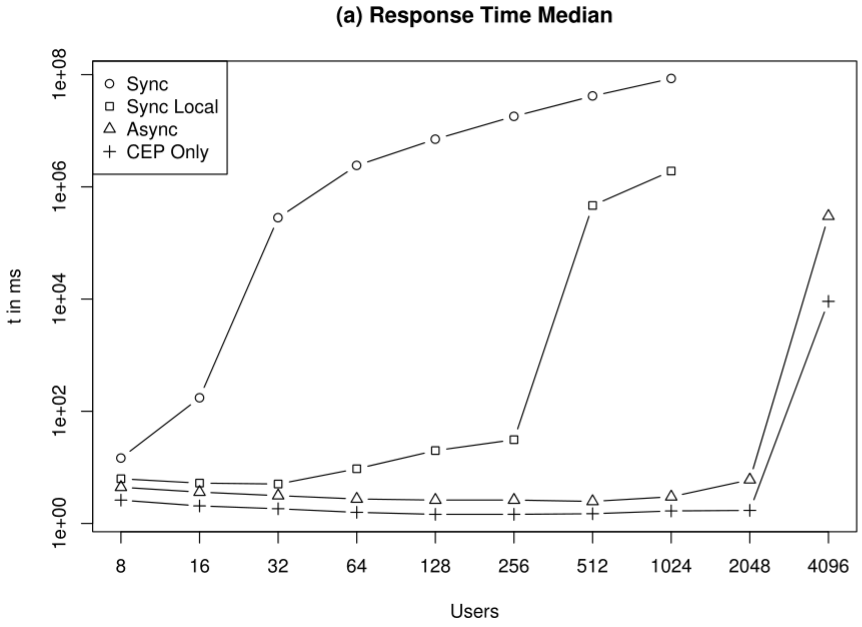
- 46 Production/ECA Rules

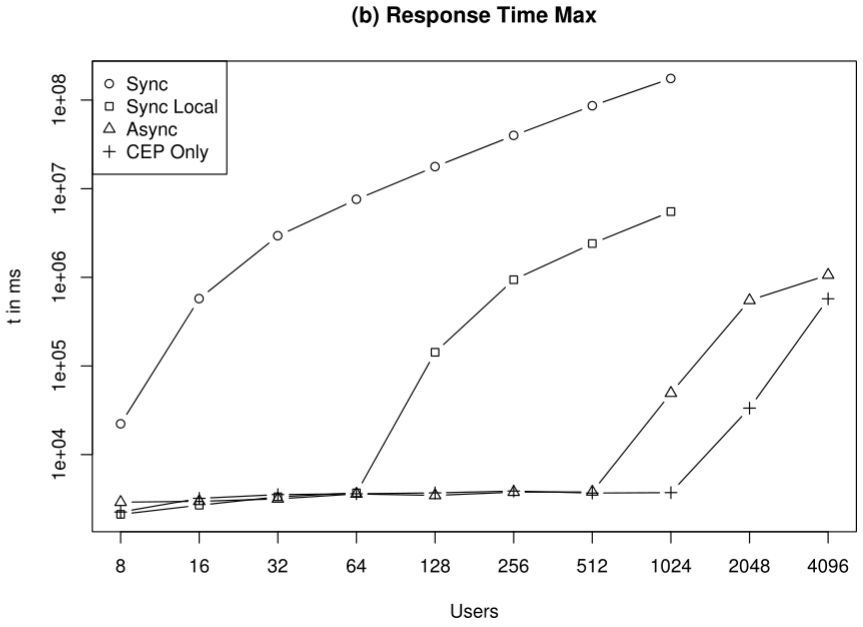
RHS \ LHS	(a)-(d)	(e)-(f)
(g)	1	0
(h)-(i)	15	30

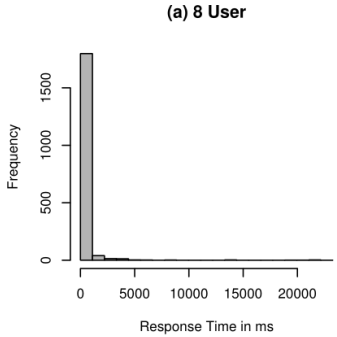
$$2^n \text{ Users} \times 0.67 \frac{\text{Events}}{\text{User} \times \text{s}} \times 300s = 201 \times 2^n \text{ Events}.$$

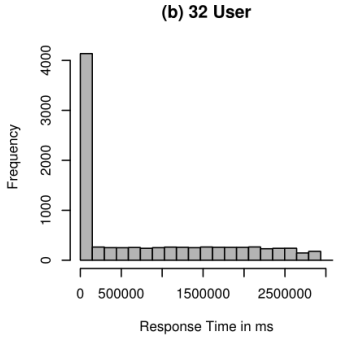
- with n = 3, …, 12 => (8, 16, …, 4096) → experimental users
- equals: $(7 * 10^5, 1.3 * 10^6, \ldots, 3.5 * 10^8)$ → “real users”

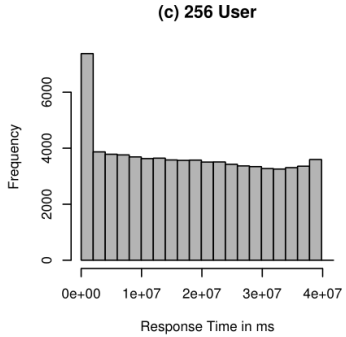
Hybrid Approach
Experimental Results

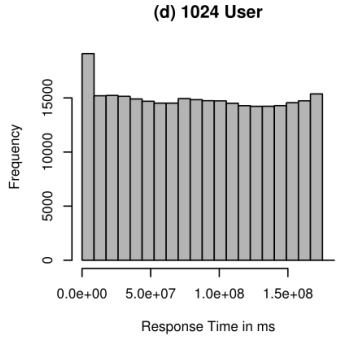




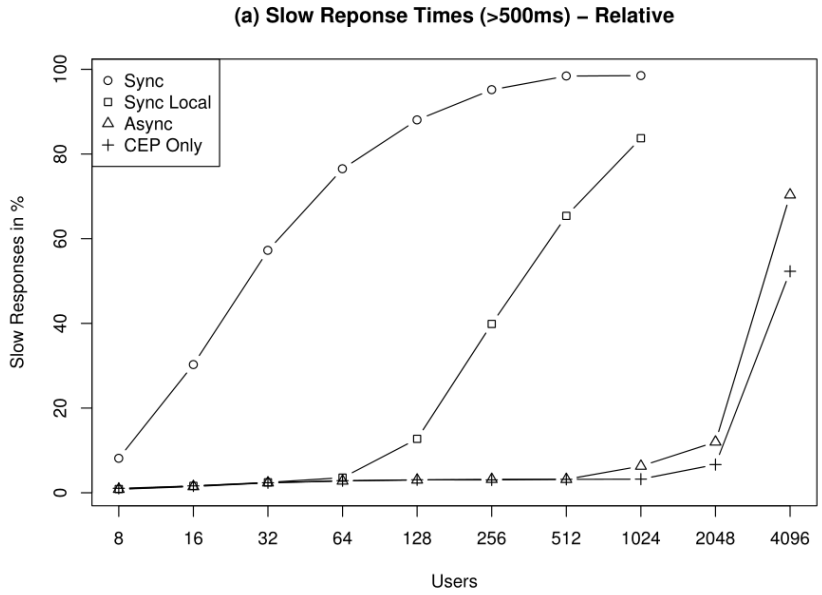


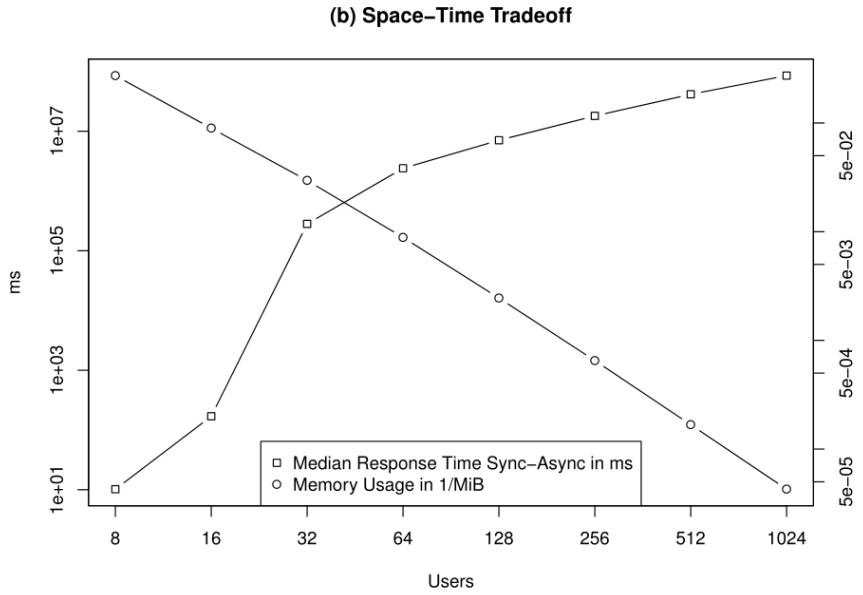






Hybrid Approach
Experimental Results





Open Challenges
Transaction Strategies

- **EPN Scenarios?**
- **Strategies**
  - **One Call per LUW**
    - API strategy (all locked)
    - High Concurrency (trade-off strategy, no read locks => stale data; dirty/phantom reads)
  - **Multiple Calls for LUW**
    - Client Orchestration (i.e., Rule Engine is Client and has to handle rollbacks)
    - High Performance (i.e, compensation manager records and rolls-back transactions)

    => rule base with all inverse operations necessary!

Thank you

Philipp Herzig
SAP AG / SAP Research
Chemnitzer Straße 48
01148 Dresden

Appendix

Existing Platforms (Overview 1)

Feature	Bunchball	Badgeville	Bidgoor	Gigya	IActionable	Punchtab	OpenBadges	UserInfuser
Type	Proprietary	Proprietary	Proprietary	Proprietary	Proprietary	Proprietary	OpenSource	OpenSource
Simple Rules	YES	YES	NO	YES	YES	NO	NO	NO
Complex Rules	NO	NO	NO	NO	NO	NO	NO	NO
UI Widgets	YES	YES (Javascript)	YES (Javascript)	YES (Javascript)	NO	NO	NO	YES (HTML+CSS)
Mobile	NO	YES	NO	NO	NO	YES	NO	NO
Analytics	YES	YES	YES	YES	YES	YES	NO	YES
Communicatio	RPC	RPC	RPC	RPC	RPC	RPC	RPC	RPC
Reactive/-Asynchronous	NO	NO	NO	NO	NO	NO	NO	
Level of Integration	API and Modules	API	Modules	API	API	Modules	API	API
Delivery Model	SaaS	SaaS	SaaS	SaaS	SaaS	SaaS	SaaS	SaaS
B2B Interaction	NO	NO	NO	NO	NO	NO	YES	NO
Pre-integration	• Jive • Salesforce • IBM Connections	• Bazaarvoice • Salesforce • Yammer • Lithium • Jive • IBM Connections • Omniture	None	None	• Salesforce	• Joomla • Blogger • Drupal • Ning • E-Mail	None	None

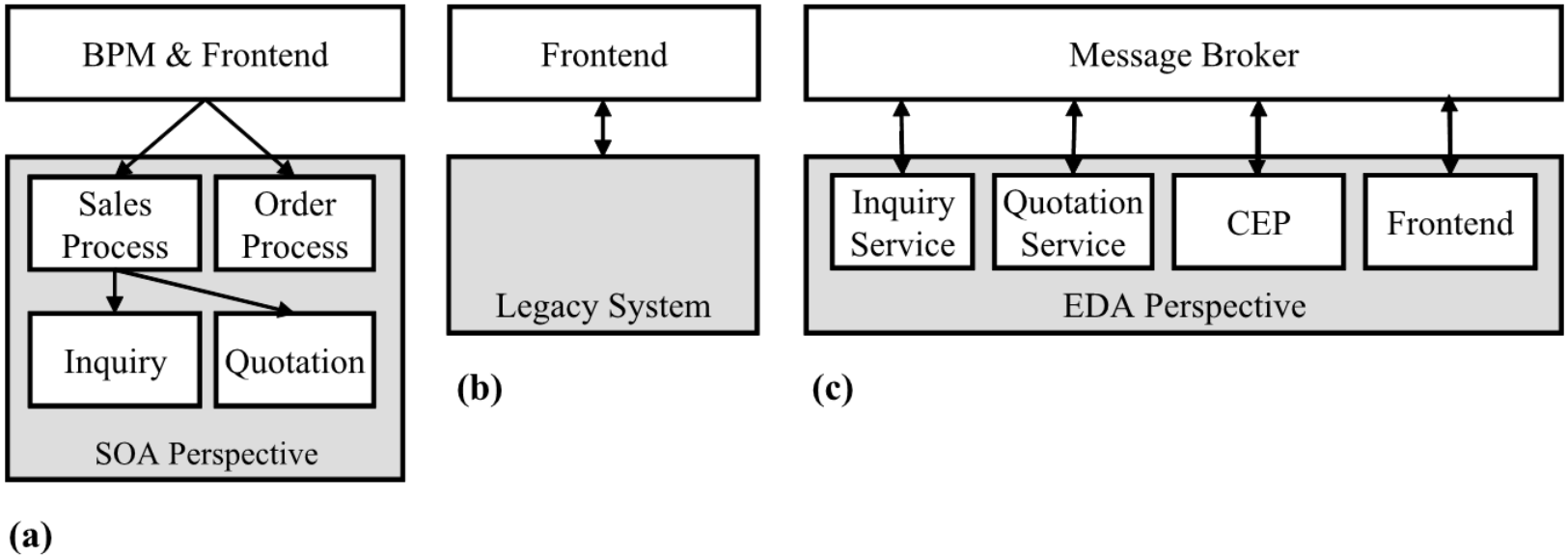
Existing Platforms (Overview 2)

Feature	Bunchball	Badgeville	Bidgoor	Gigya	IActionable	Punchtab	OpenBadges	UserInfuser
Game Mechanics	• Challenges-/Missions • Trophies • Badges • Achievements • Standard Points • Redeemable Points • Player Levels • Leaderboards • Avatar • Virtual goods • Virtual rooms • Team Competitions • Social Network Integration	• Standard Points • Redeemable Points • Player Levels • Badges • Challenges-/Missions • Social Network Integration	• Virtual Currency • Badges • Player Levels	• Single non-redeemable point metric • Player Levels • Leaderboard	• Points • Leaderboard • Badges • Levels	• Leaderboard • Single reedemable point metric • Badges • Giveaways • Social Network Integration • Activity feed	• Badges	• Points • Badges • Leaderboards

Gamification Platform
Example Rule

```
rule "newUser"
  when
    $evt : EventObject(type=="new_user") from
         entry-point eventstream
  then
    GamificationEvent ge = new GamificationEvent();
    ge.setType("createPlayer");
    ge.setPlayerid($evt.getPlayerId());
    ge.put("logonname", $evt.get("logonname"));
    bus.publish(ge);
    retract($evt);
end
```

Gamification Platform
Existing Types



- **Frontend Integration**
  - Support for various platform and technologies
  - Generic generation of UI widgets
  - Current solutions are limited to the generation of HTML5 „components“
- **Analytics**
  - Analytical support the monitoring phase of the gamification cycle
  - Simple measures
    - Revisits
    - Participation rate
  - Complex measures (Higher-order constructs)
    - Engagement Levels
    - Game-theoretical algorithms (estimate payoff matrices)

Related Work

- **OpenSource**
  - UserInfuser
  - OpenBadges
- **General Purpose Platforms (Closed Source)**
  - Badgeville
  - Bunchball
- **Specific Platform (Closed Source)**
  - IActionable
  - Gigya
  - Bigdoor
  - Punchtab
  - etc…

Current Approaches

- **Open source**
  - Simple game mechanics programmatically
  - Remote storage for game mechanics
  - Translation of traditional achievement systems, e.g., Xbox Live, iOS GameCenter

- **Closed source space**
  - Hardly documentation available
  - Striking marketing promises
  - However projects report large integration projects
  - Tight coupling of application with gamification platform
  - Data Silos
  - SaaS offerings
  - Strong focus on consumer and web applications