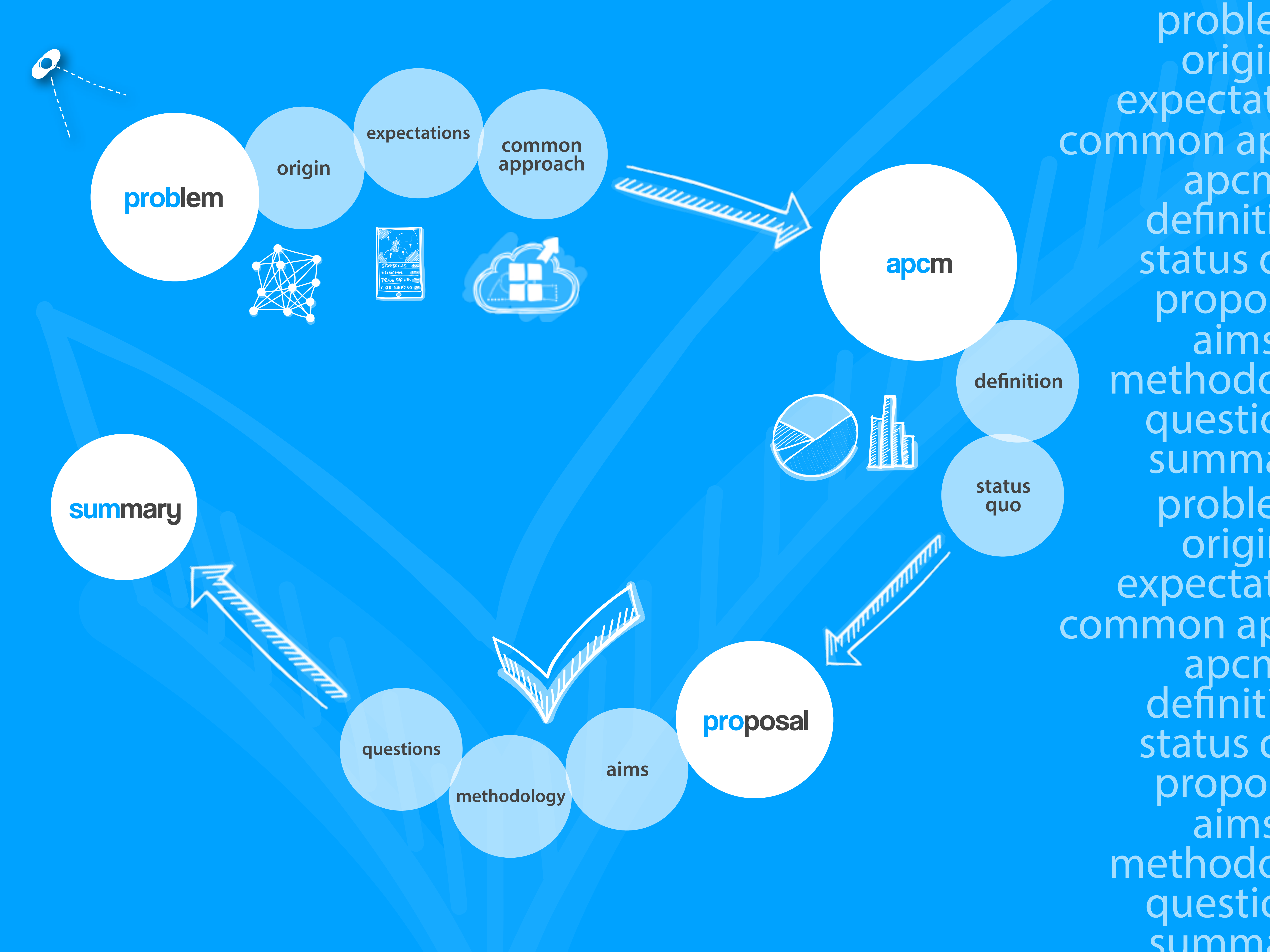
apcm

The Asynchronous Persistent Cache Model

Performance Evaluation in the Web Context

UWS UNIVERSITY OF THE WEST of SCOTLAND

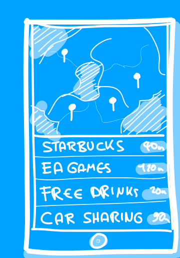
PhD Research Subject Pitch by Thomas Fankhauser


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary

theProblem: origin

The combination of the **Social Web**, the increase of **Mobile Smart Devices** and **Cloud Hosting** change the requirements for **Web Applications**

Social Web



Mobile Smart Devices

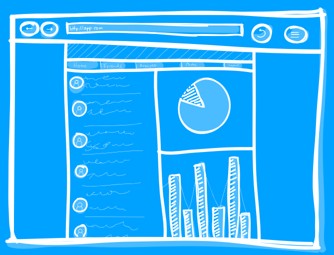
Cloud Hosting

Lots of Content

Lots of Requests

Scaleable Infrastructure

theProblem: expectations



Web Application

Handle 10k+ Concurrent Requests

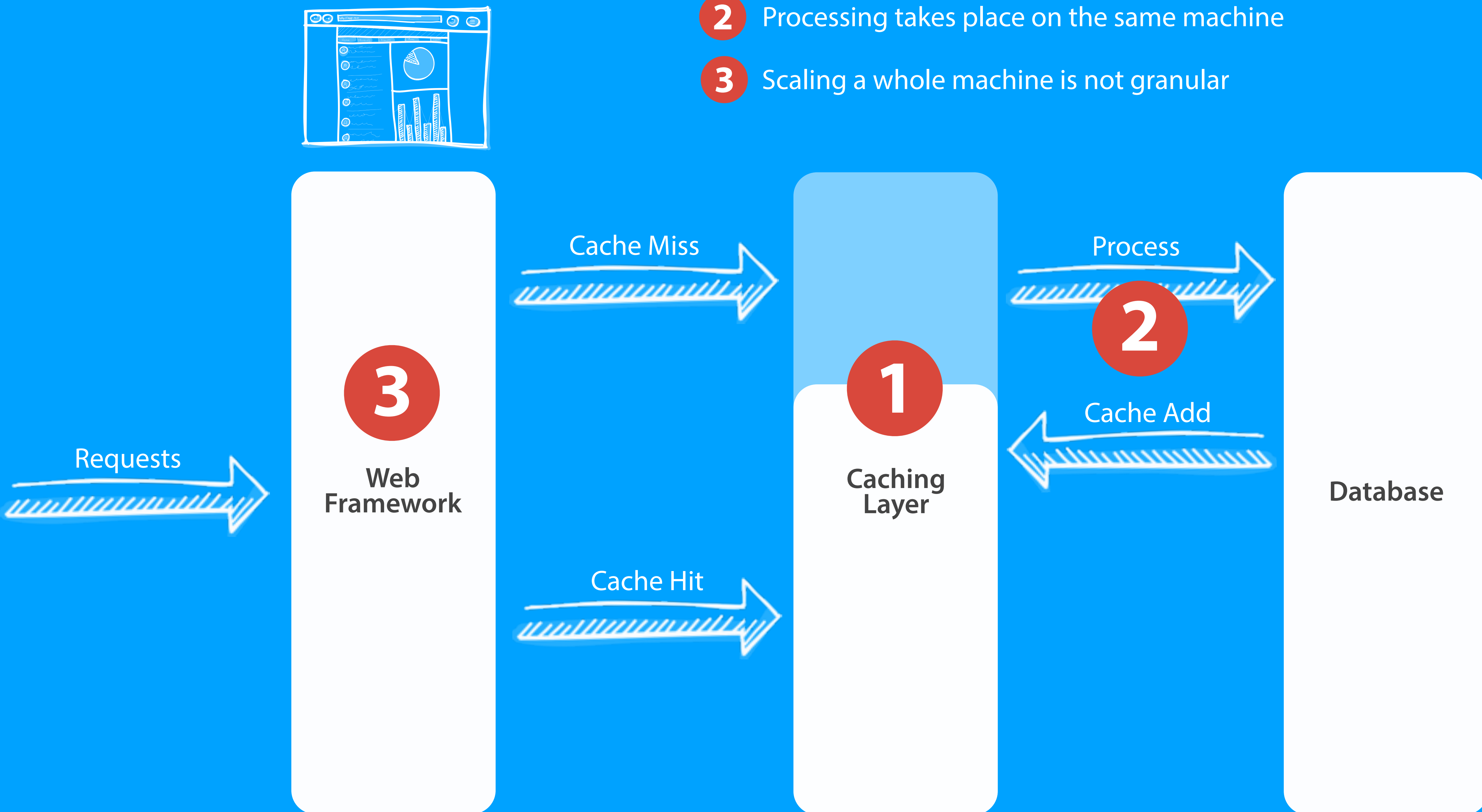
Requested Content has a High Variance

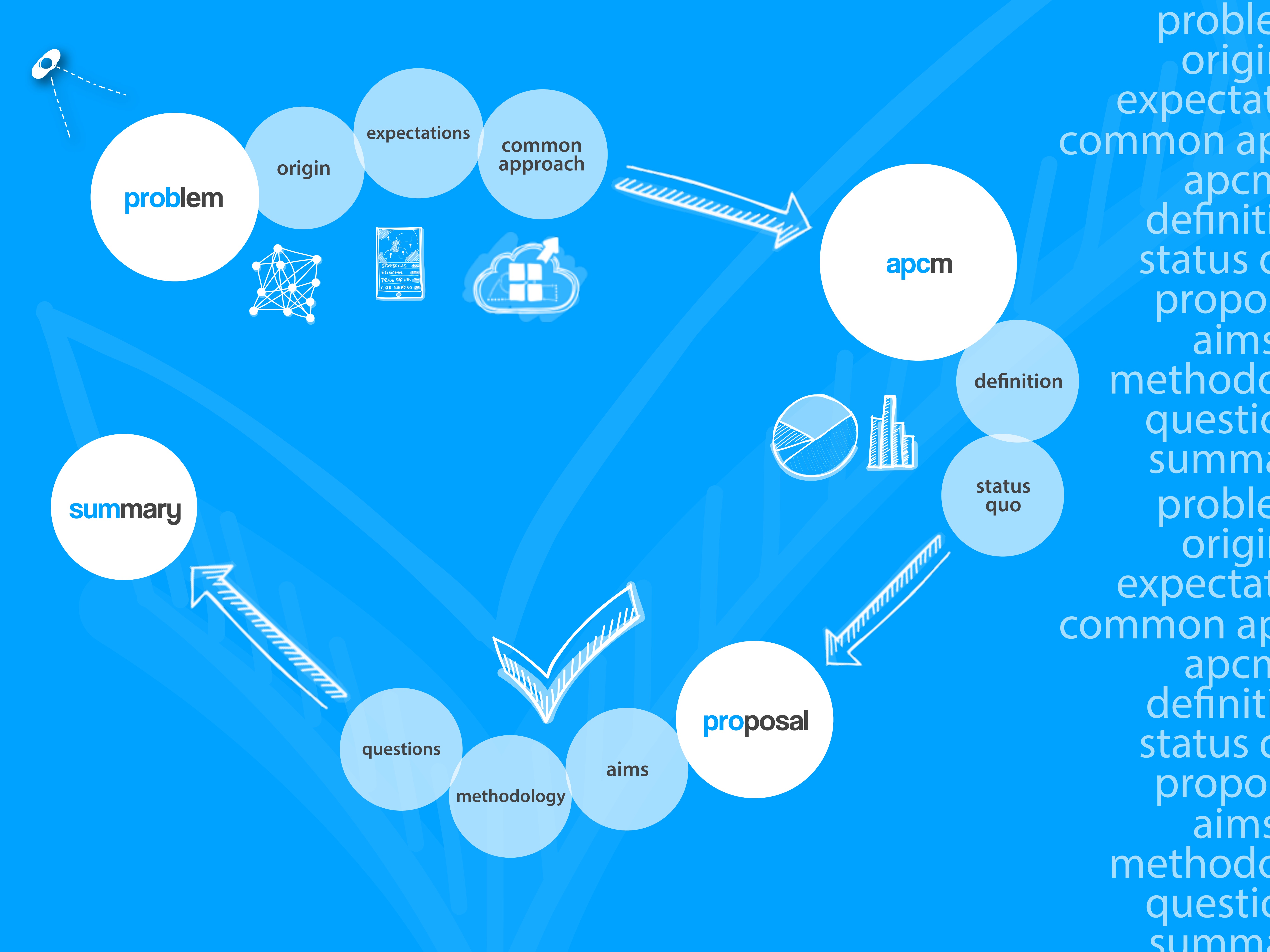
User expects Low Latency and Guaranteed Response

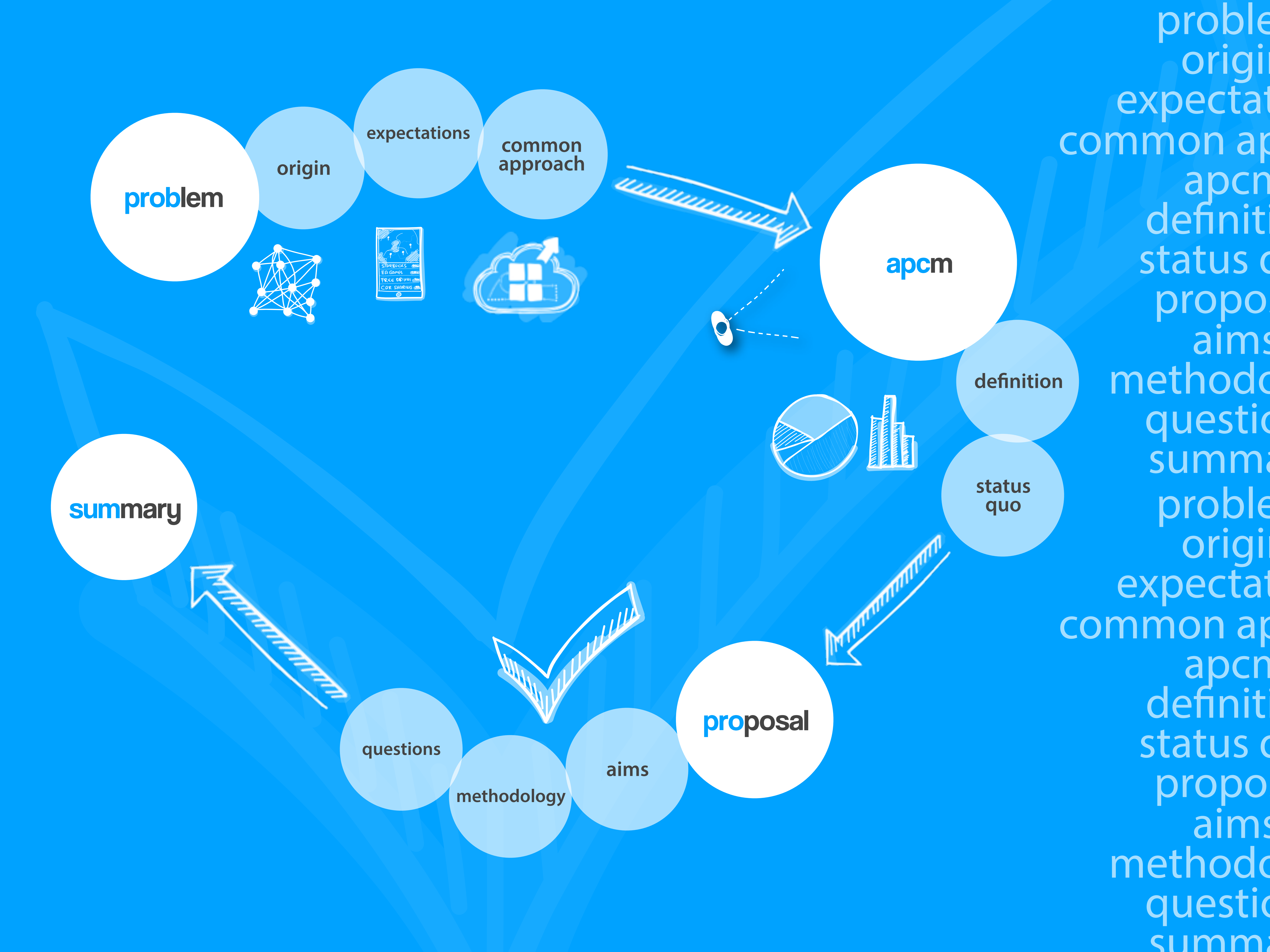
Scaleable Architecture with Maximum Resource Utilization

theProblem: common approach

1. When and why is cache invalidated?
2. Processing takes place on the same machine
3. Scaling a whole machine is not granular

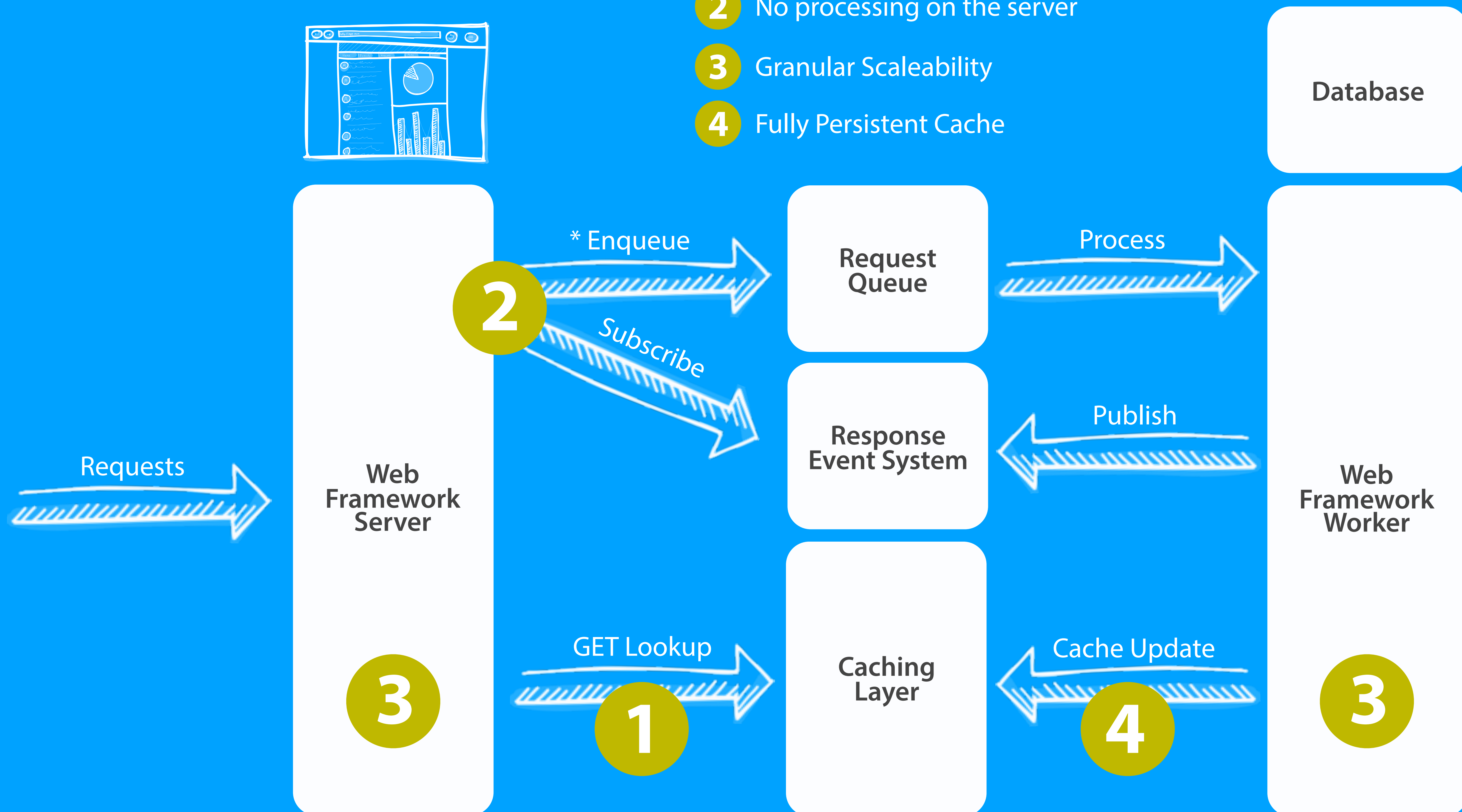



problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary

theAPCM [1]

1. No cache misses and invalidation
2. No processing on the server
3. Granular Scaleability
4. Fully Persistent Cache



theAPCM: definition

Asynchronous → Enqueue and Subscribe is non-blocking

Persistent → All content is in a valuable cache - no fallbacks

Cache → All content is ready to deliver - anytime

Model → There are various possible implementations for the concept

theAPCM: status quo

Proof of concept implementation **Scales [2]** could not render the concept useless:

Scales was able to deliver between 1000% and 6000% more requests per second then an uncached reference implementation

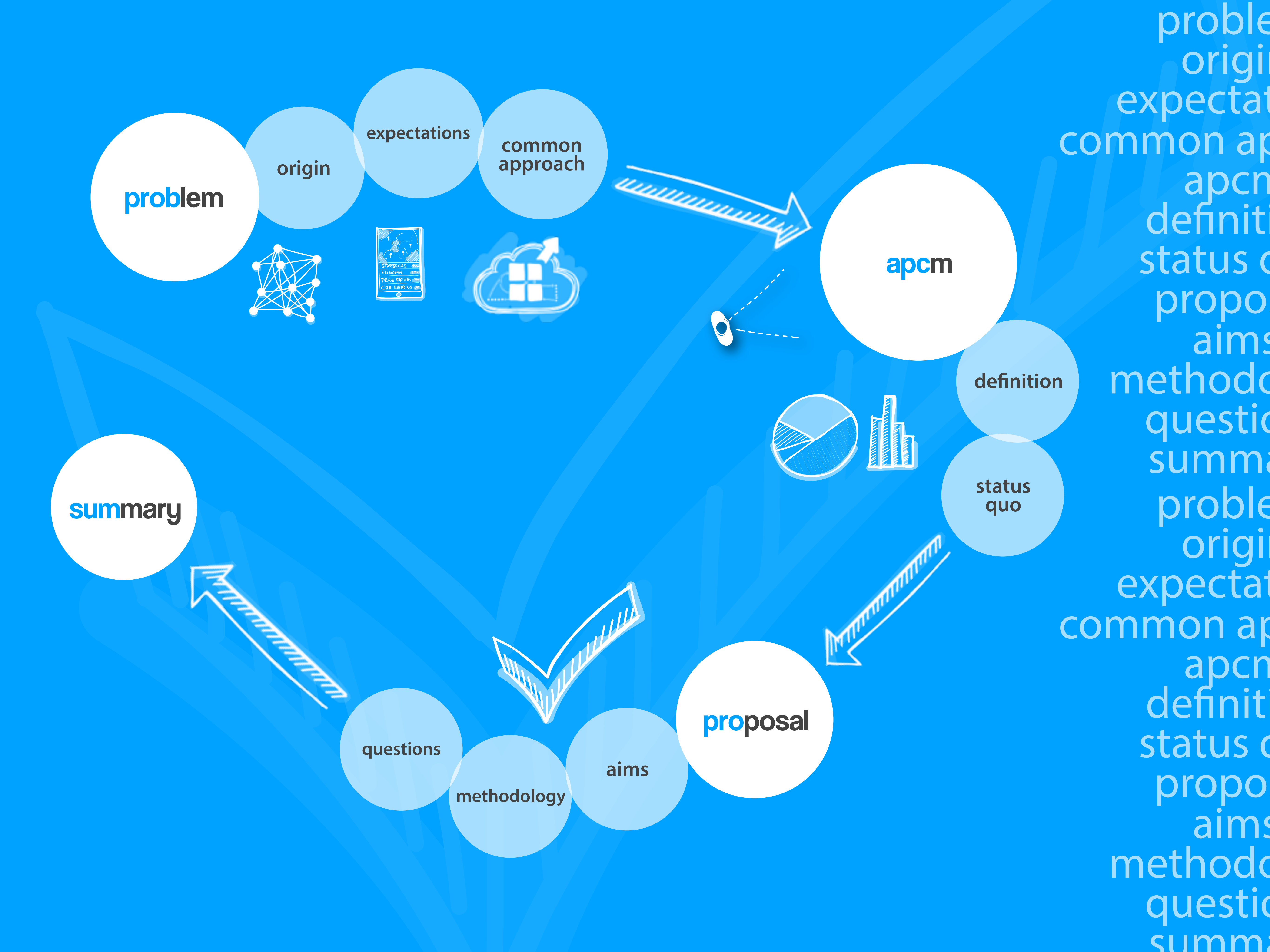
but

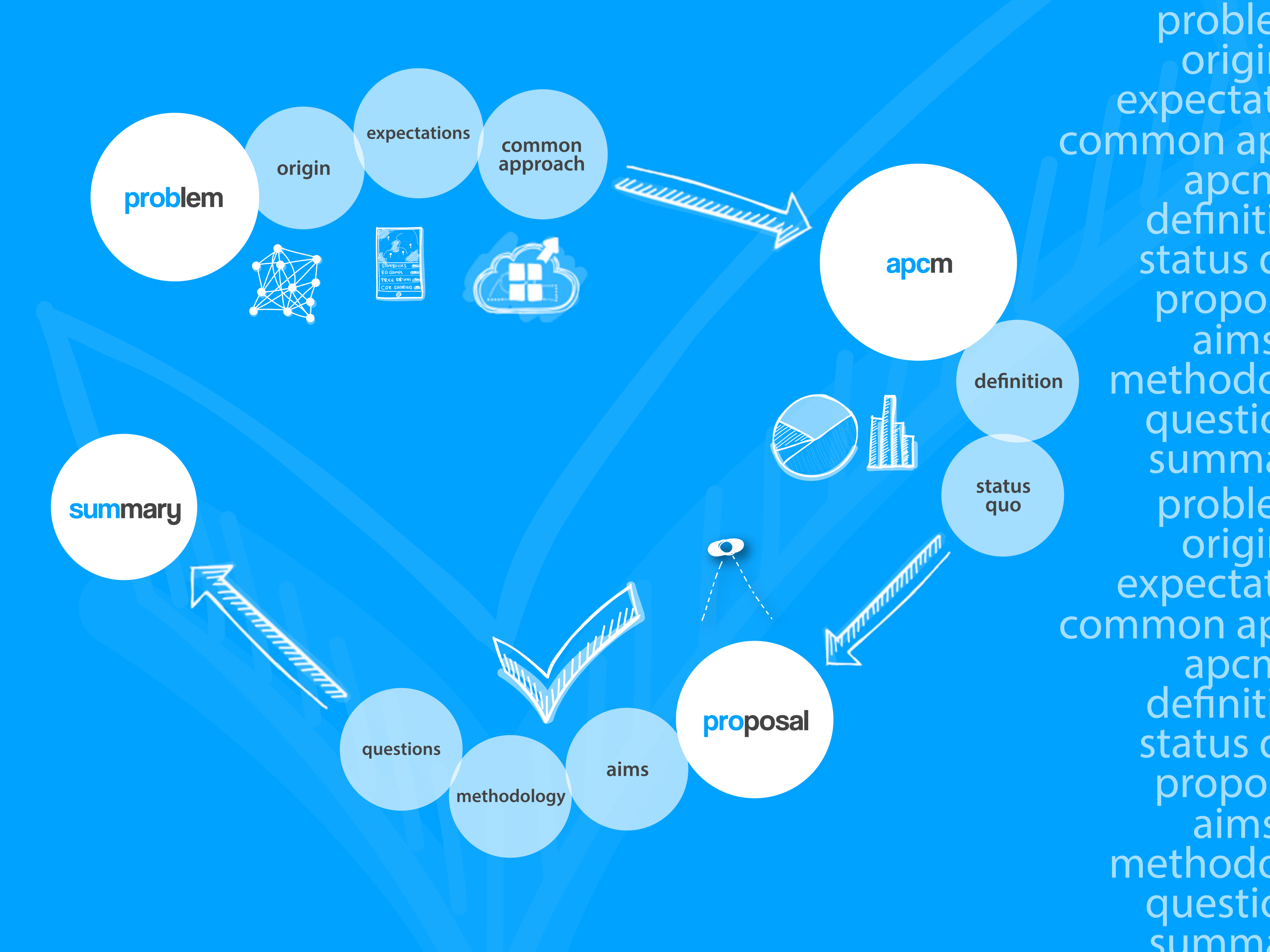
There is no data that researches a real-world application using the concept

The lessons learned from Scales allow a better implementation

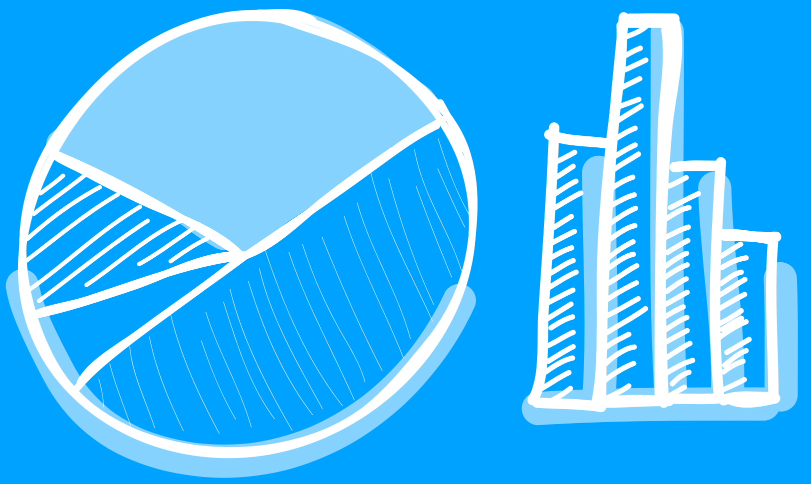
There is no data that researches the development overhead on the application side

There is no data that researches the behavior under high load conditions


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary

theProposal: aims



Evaluate the performance of the APCM with collected data to be able to make a meaningful statement about its performance or show why and when the concept isn't better than current approaches

theProposal: methodology

1 Abstract the proof of concept to a formal model

Formalized Model

2 Create a generalized implementation

General Implementation

3 Collect real-world application performance data in different scenarios and compare it to the original application performance data

Performance Data

4 Evaluate performance hypothesizes and examine bottlenecks for a failure of concept or implementation

theProposal: questions

How is the utilization of a minimal setup compared to a traditional approach?

Is there a load-barrier where the pre-computing gets exponentially expensive in terms of cache-updates?

Is it applicable to maintain an index of all available resources during development?

Is the APCM applicable for all sizes of applications?

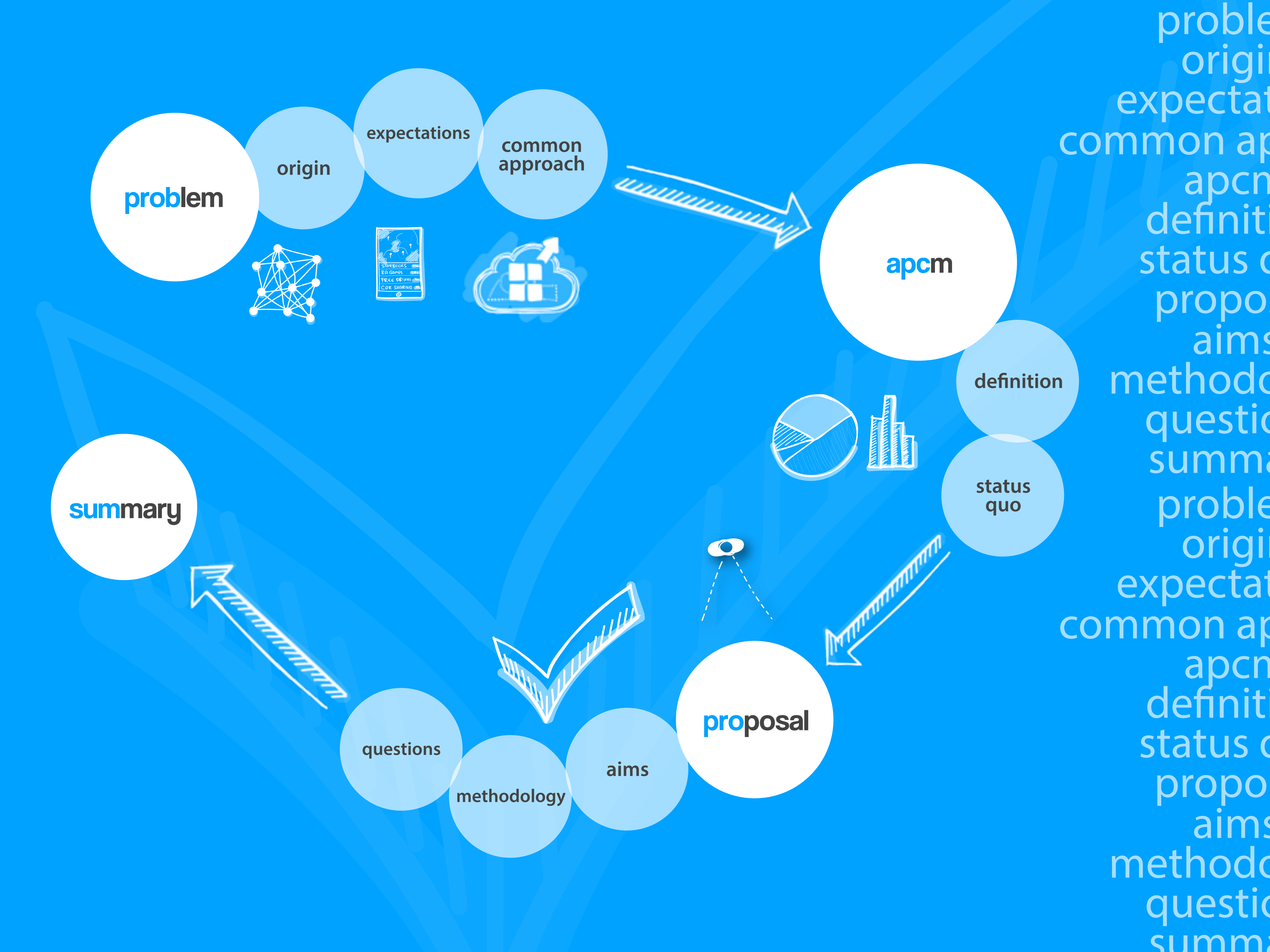
How is the resource utilization of the APCM compared to traditional, manual caching approaches?

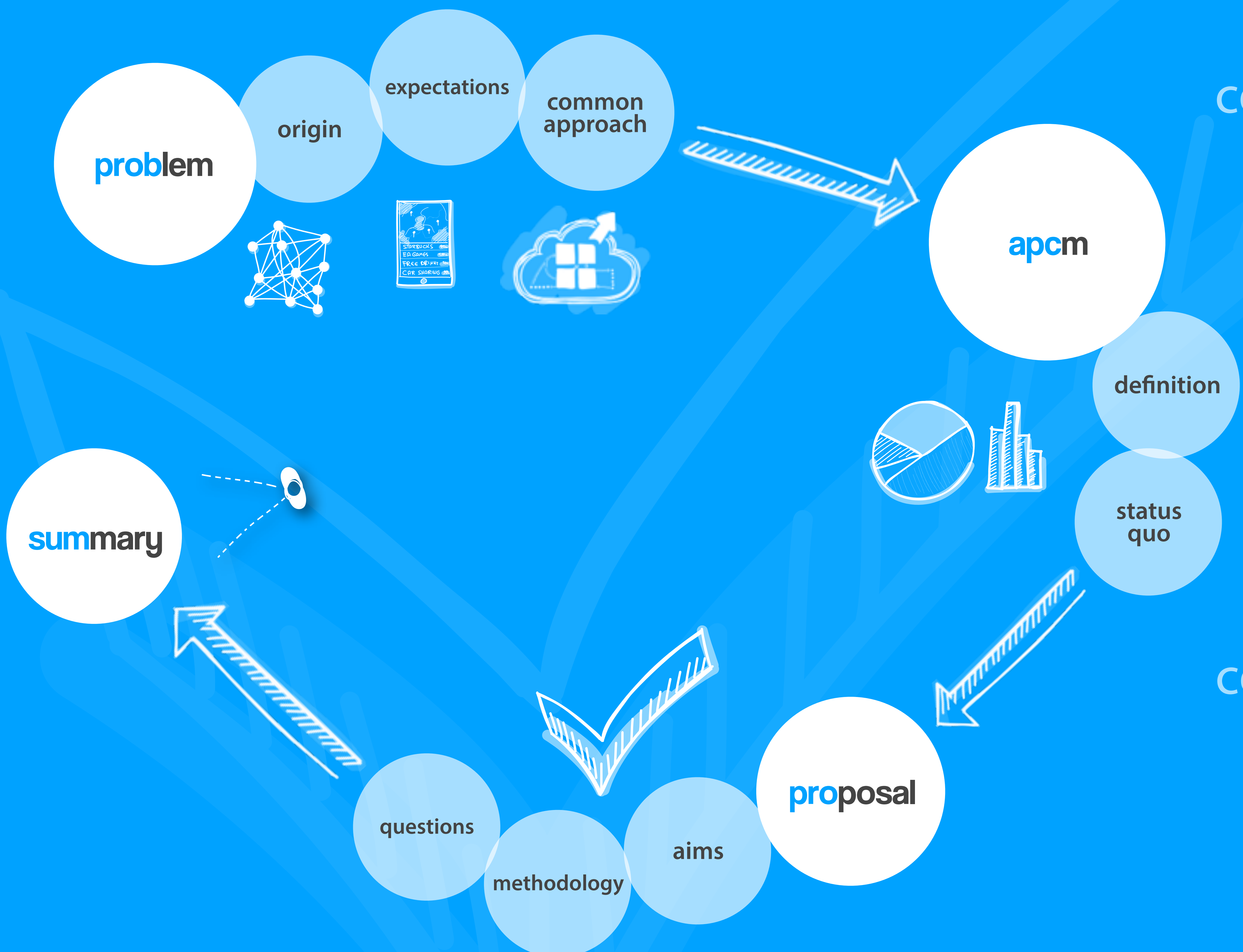
Where are the points for measuring the system performance?

How does the model scale in terms of agile development? How are changes published through the system?

How brittle is the balance of the model? Are there drift effects? Are there storms?

How does the logic work that determines the adding or removal of server-, worker-, cache-, event- or queue-nodes?


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary


problem
origin
expectations
common approach
apcm
definition
status quo
proposal
aims
methodology
questions
summary

theSummary

Why?

Web Applications need to support lots of content and requests in the future. Resources are available from the cloud.

What?

The APCM could be a better way to build such high-demanding Web Applications. But until there isn't a collection of real-world performance data it is just a concept.

And then?

If the research renders the concept as performant, lots of implementations on different platforms could be made helping developers to build scaling applications by default. The results could have a huge impact on the resource utilization and thereby also reduce the general energy consumption of an application.

The Asynchronous Persistent Cache Model

Performance Evaluation in the Web Context

PhD Research Subject Pitch
Thomas Fankhauser, M.Sc
tommy@southdesign.de
+49 16094988551

References

[1] Thomas Fankhauser, 2012. Super Scale Systems, Media University, Stuttgart
[2] Scales project is derived from [1] and available at: http://itscales.org