

QualityConf 2011



Evolution: Von Performance Tests zur produktiven Anwendungsüberwachung

Manuel Núñez, Stefan Ruppert
MyARM GmbH
Altkönigstraße 7
65830 Kriftel
Deutschland
web: <http://www.myarm.com>
eMail: info@myarm.com

Inhalt

- Annahmen und Motivation
- Modellbildung
- Application Response Measurement Standard (ARM)
- Anwendungsinstrumentierung
- Testzyklus
- Integration in produktive Umgebungen
- Kennzahlen

Annahmen

- Verteilte Systeme sind gutes Beispiel für relevante Szenarien
- Nonfunktionale Aspekte zu oft vernachlässigt
- Messen „realer“ Werte ist unersetzlich
- Testphase hat bei Messungen herausragende Bedeutung
- Gemeinsam genutztes Werkzeug notwendig

Motivation (nonfunctional)

- Nonfunktionale Aspekte sind schwer zu erfassen
 - Hohe Komplexität Verteilter Systeme
 - Fehlende Instrumentierung
 - Fehlende Lasttreiber
 - Infrastruktur anders als Produktionsumgebung
 - Manche Probleme sieht man nur im Betrieb

Motivation (Messungen)

- Im Ingenieurwesen werden Systeme durch *Messungen* über die gesamte Lebensspanne qualitätsgesichert
 - Materialien (Dichte, Härte, Leitfähigkeit)
 - Prozesse (Schweißstrom, Pressdruck)
 - Funktionsprüfungen (Airbag)
 - Betriebswerte (Öltemperatur, Füllstände)
 - Fehleranalysen

Modellbildung (Messgrößen)

- Interessante Kriterien für Softwaresysteme
 - CPU-Zeitverbrauch
 - Netzwerklatenzen, Bandbreiten
 - Speicherverbrauch
 - Fachliche Kennzahlen
- Alle relevanten Messwerte schlagen sich in der Antwortzeit nieder

Modellbildung (Antwortzeit)

- In Verteilten Systemen ist *Antwortzeit* ein sehr gutes Maß für die Qualität des Services (im Betrieb). Sie beinhaltet:
 - Netzlaufzeiten
 - Zeiten in anderen Systemen
 - Technisch bedingte Verzögerungen
 - CPU
 - Netzlast
 - Parallelisierungsgrad

Modellbildung (business tx.)

- Laufzeitmessungen als Grundlage für
 - Beurteilung eines Systems im Betrieb
 - Änderungshinweise für Architektur
 - Änderungshinweise für Entwicklung
- Auswahl der relevanten Services, Transaktionen etc. der Anwendung ist die eigentliche Modellierung
 - *business transactions*

Modellbildung (Erwartung)

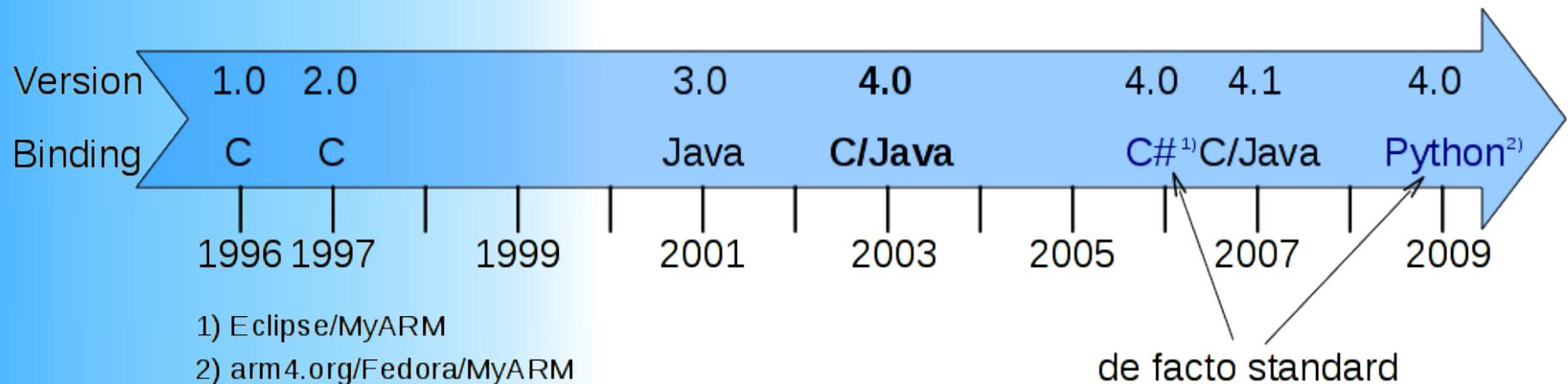
- Architektur und Entwicklung:
 - Erwartungswerte formulieren
 - Toleranzbereiche festlegen
 - Wenn bekannt, ansonsten können Informationen in Testphase abgeschätzt werden

Modellbildung (Entwicklung)

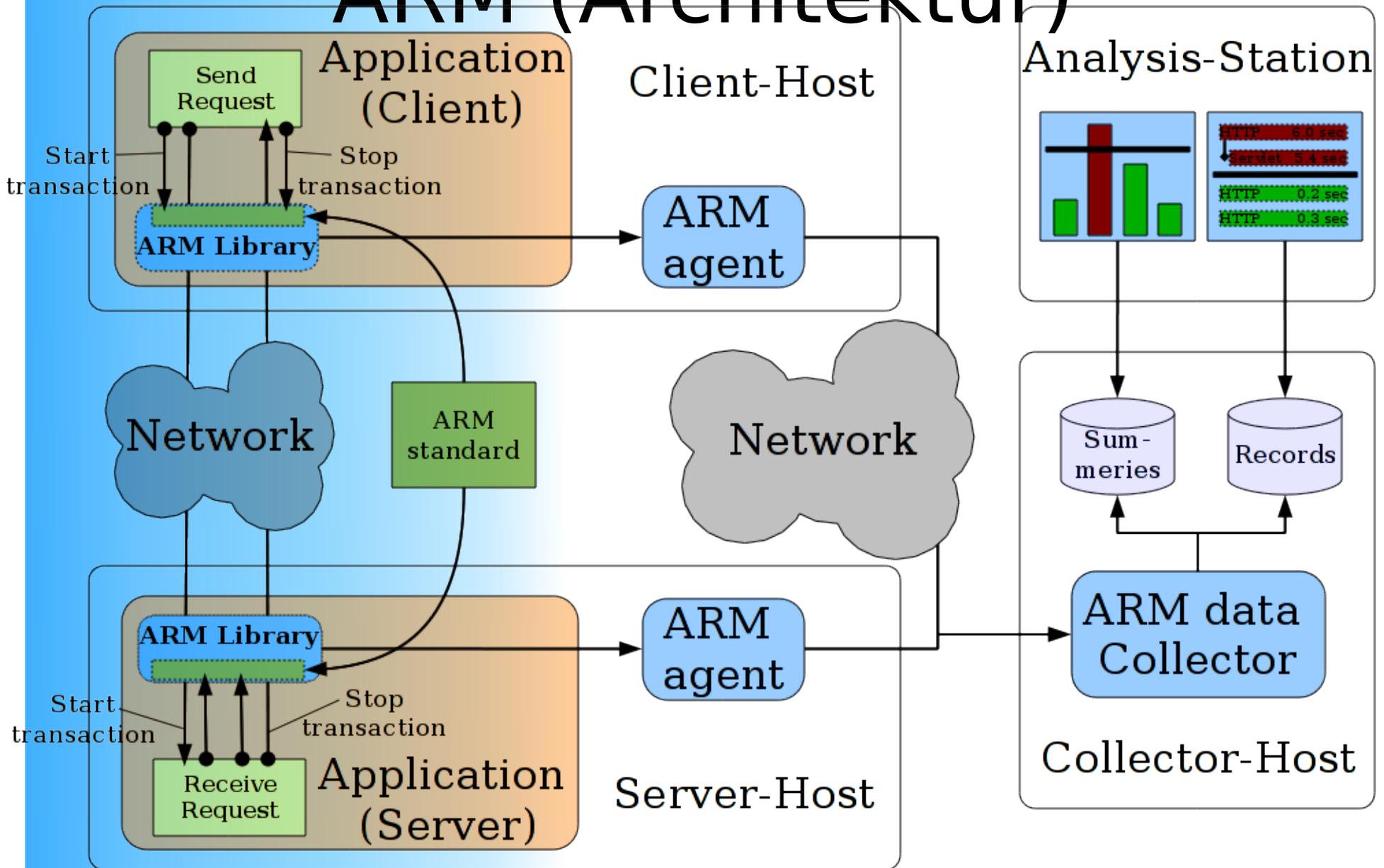
- Reichert Architektur Performancemodell um technische Messpunkte an (*technical transaction*)
- Validiert die Erwartungen
- Zeigt Engpässe auf

ARM Standard

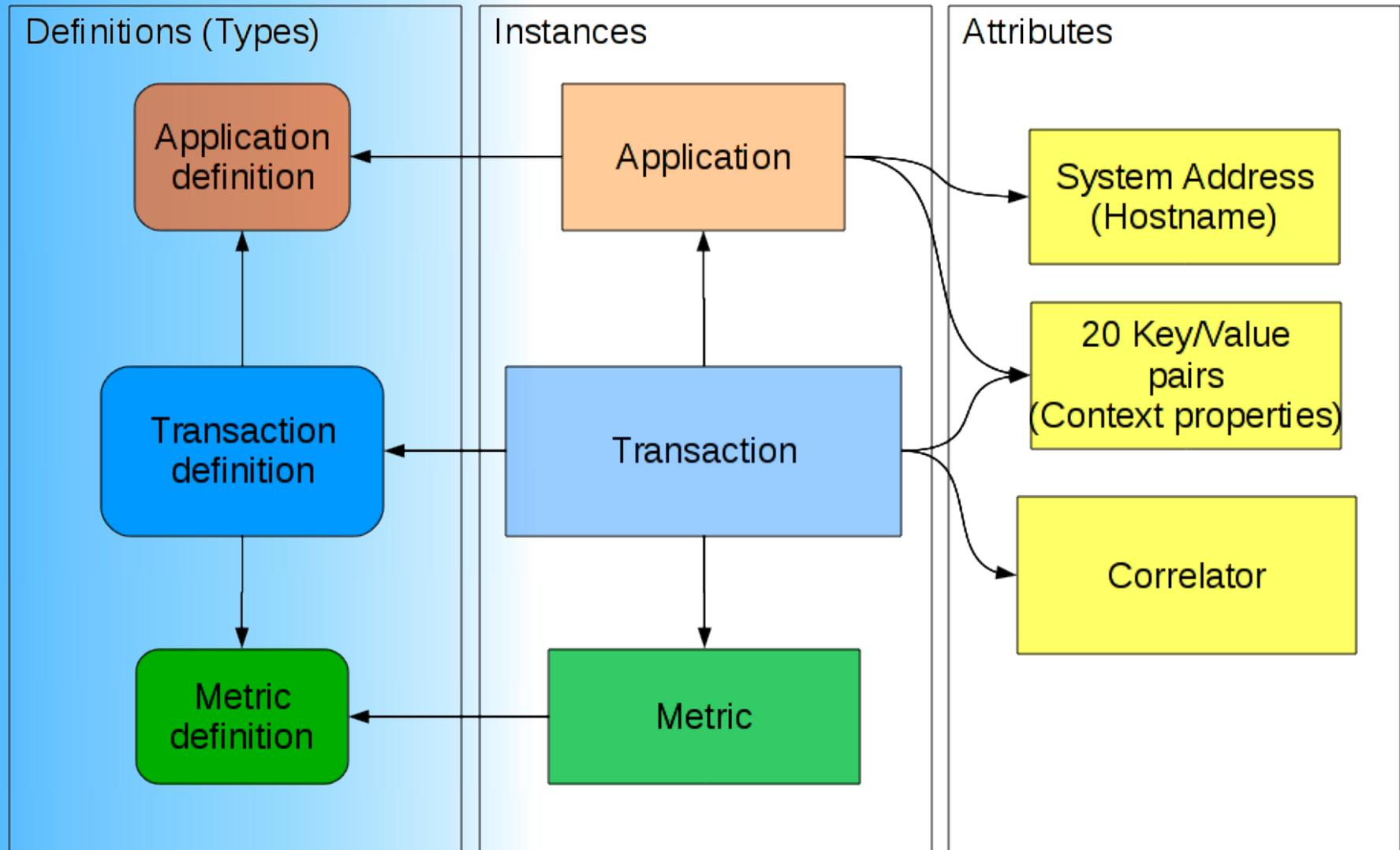
Application Response Measurement (ARM) History



ARM (Architektur)



ARM Konzepte (Überblick)



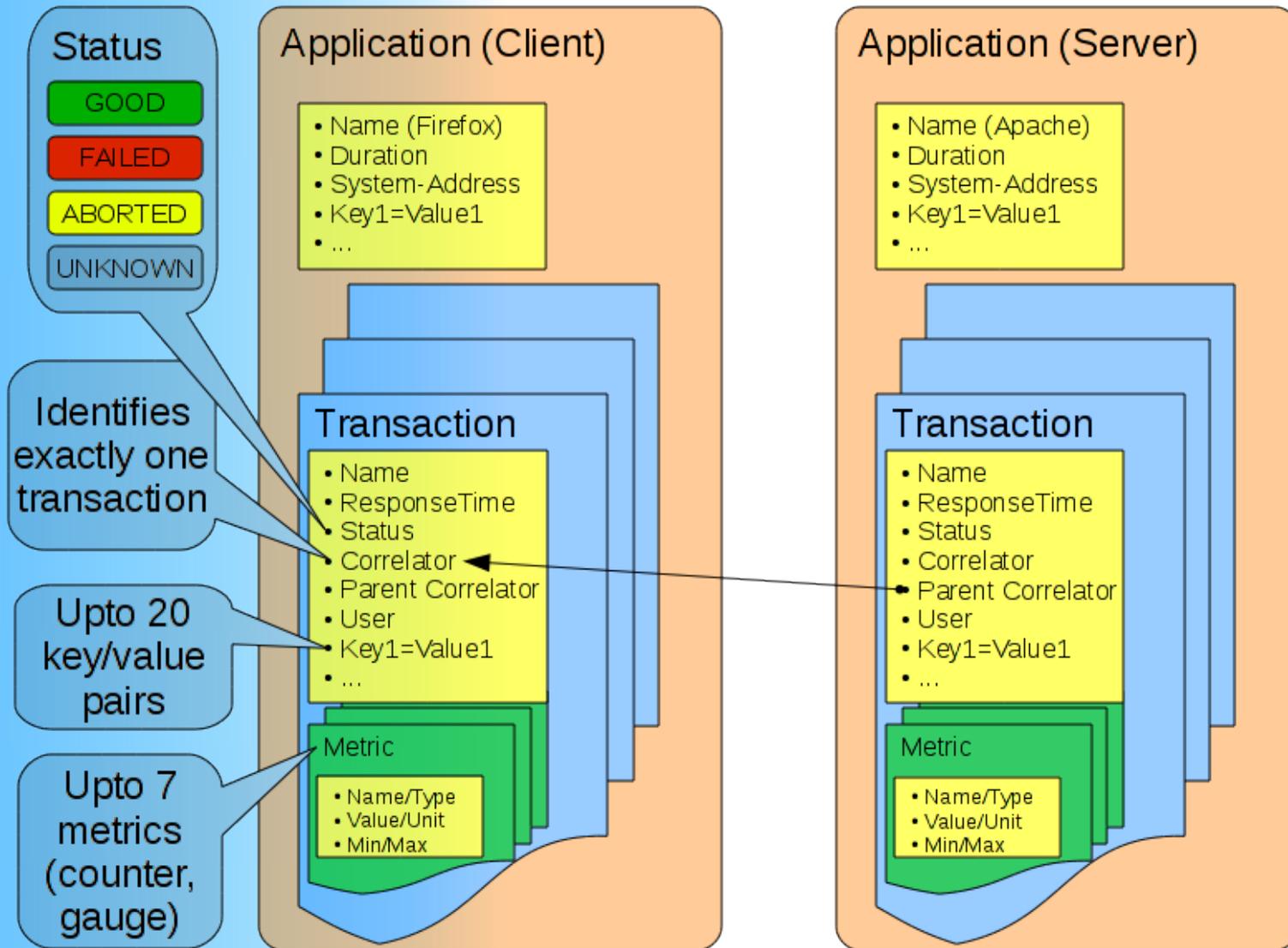
ARM Transaktionen

- Zentraler Begriff der *Transaktion*
 - Start- und Stoppzeiten
 - Definition („was“)
 - Vater-Transaktion
 - Applikation
 - Status (GOOD, FAILED, ABORTED)
 - Weitere Kontextdaten

ARM Selektionskriterien

- Kriterien, nach denen in Auswertungen selektiert werden kann:
 - Properties
 - Metrics
 - System
 - User
 - Application

ARM Konzepte (Details)



Testzyklus

- Während Entwicklungsphase messen, verifizieren etc.
- Test kann inhaltlich prüfen (entsprechen Abläufe fachlichen Vorgaben?)
- Test prüft auch Erwartungen bzgl. Performance, Fehlerbehandlung
- Ergebnisse fließen in Architektur, Entwicklung zurück

Anwendungsüberwachung

- Ermittlung von KPIs möglich
- Tracing für Fehleranalyse
- Statistische Aggregation
- APDEX-Index zur einfachen Ermittlung von tolerierbaren Zeitdauern
- Ggf. Korrelation mit Log-Einträgen
- SLAs

Kennzahlen (MyARM)

- Call-Overhead für Java typisch 2,5 - 6 μ s
- Call-Overhead für C typisch 1,1 - 4,8 μ s
- Ca. 100 Byte je Messung serialisiert
- 12000-15000 Inserts je Sekunde in DB
- Ca. 200000 Messungen/s in Applikation
- Implementierung entkoppelt von Applikation, Speicher-Backend
 - Keine Störung der Applikation

Zusammenfassung

- Instrumentierung für alle beteiligten Parteien nützlich
- Nonfunktionale Aspekte lassen sich einfach messen, überprüfen
 - Einbautes Konzept der Korrelatoren
- Verteilte Architektur dabei kein Problem
- ARM gestattet auch Messung funktionaler Kennzahlen

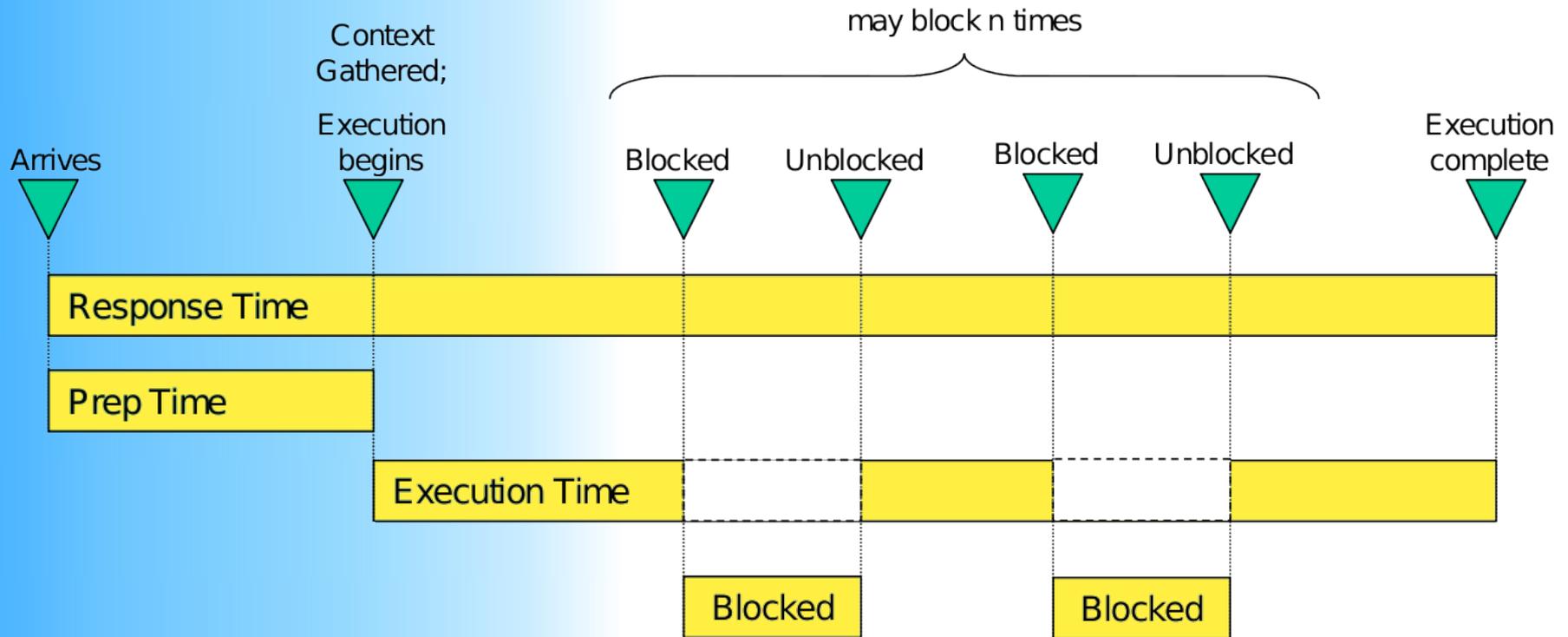
Ende

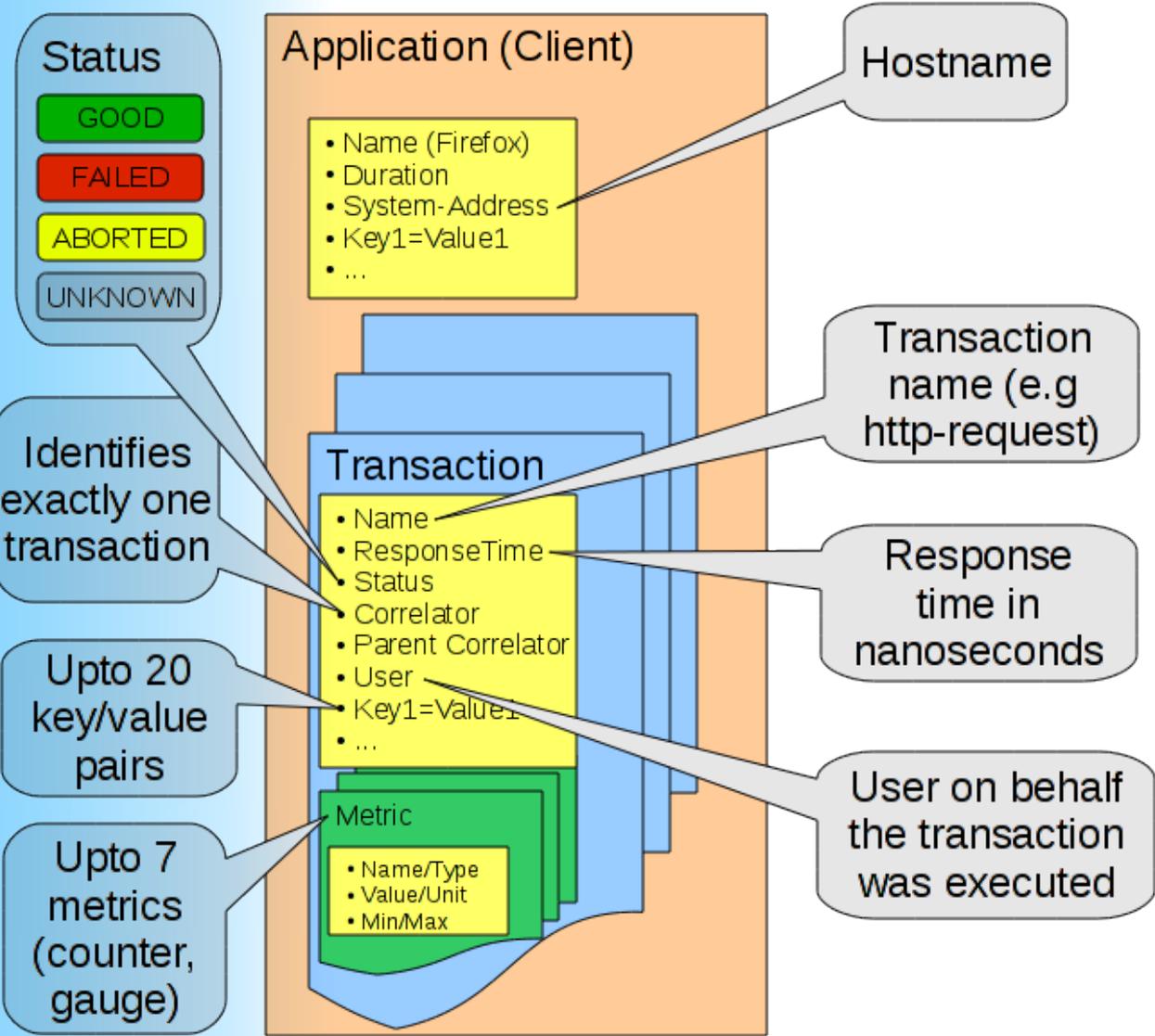
Danke für Ihre Aufmerksamkeit!

Background

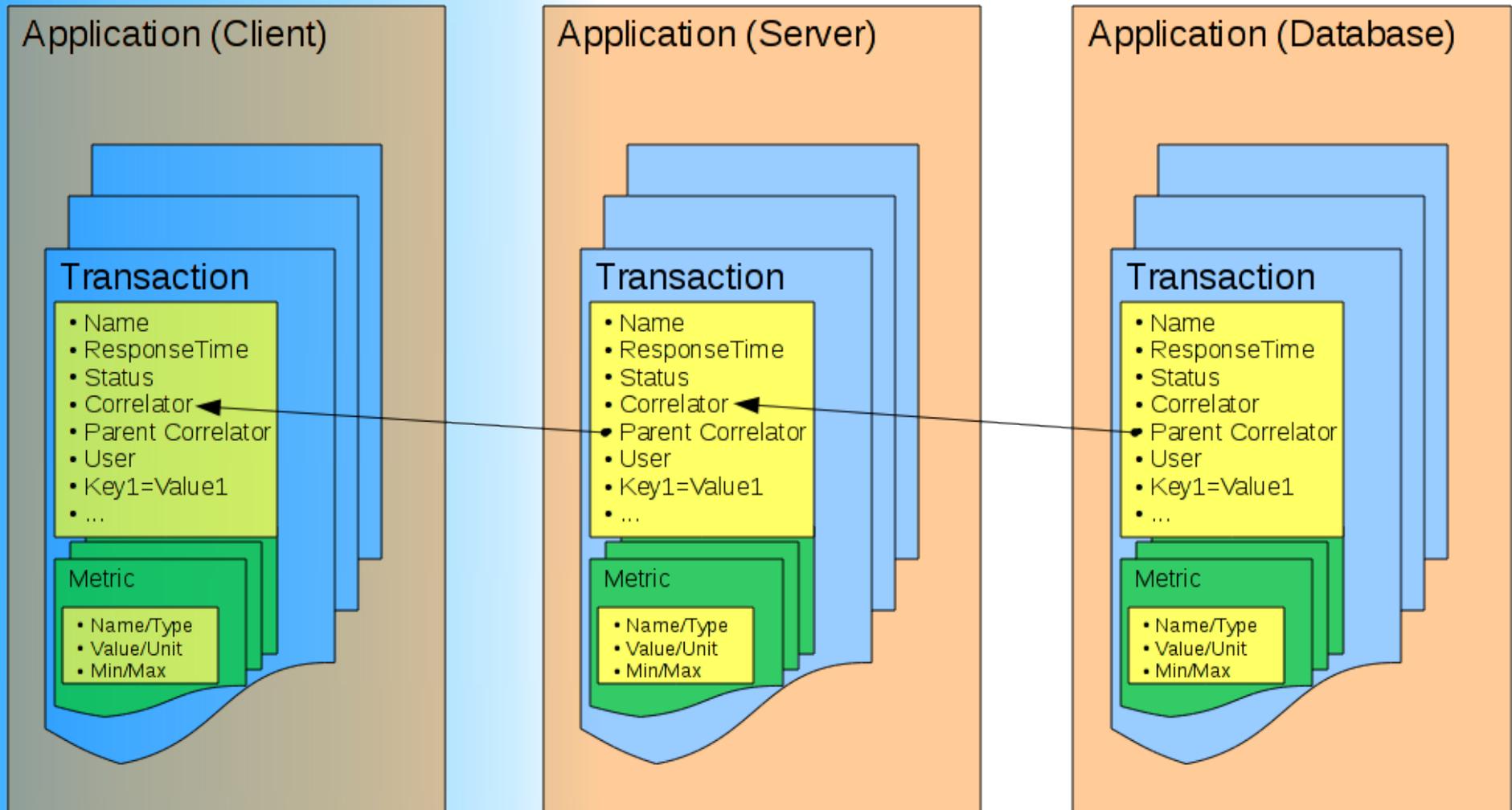
ARM Architektur

ARM (response time)

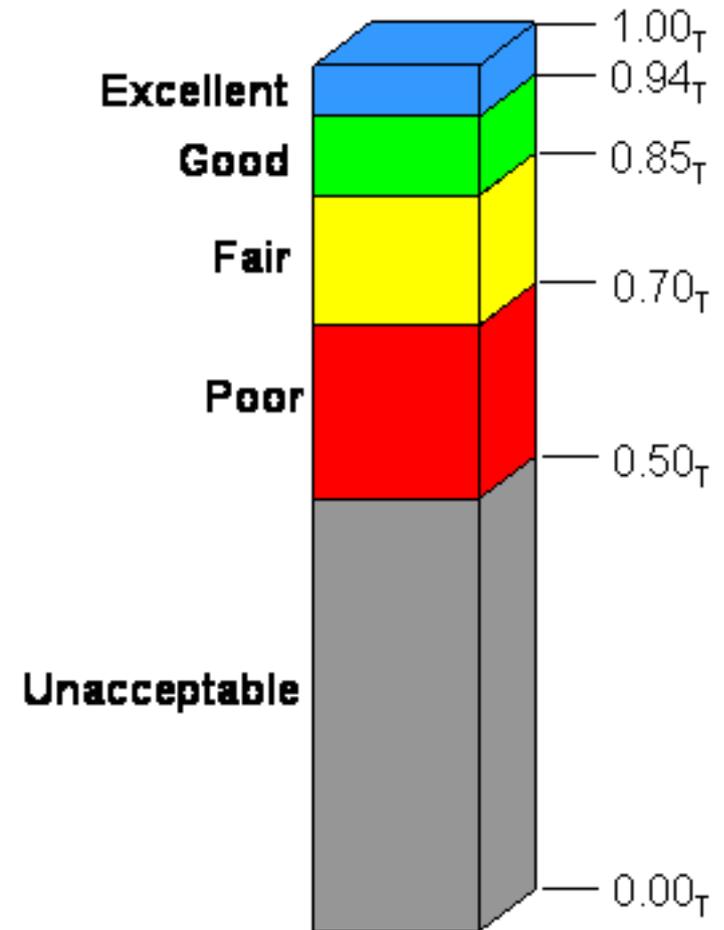
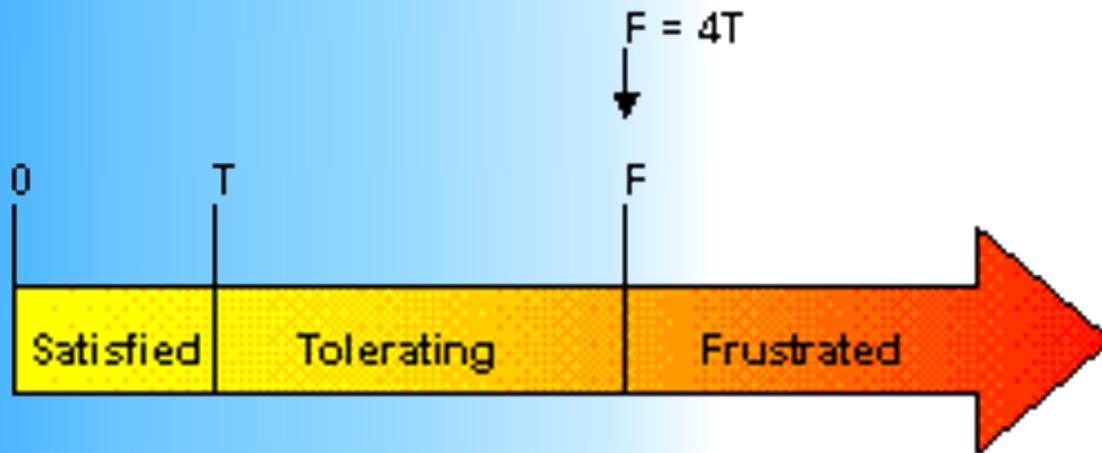




ARM Korrelationsbeispiel

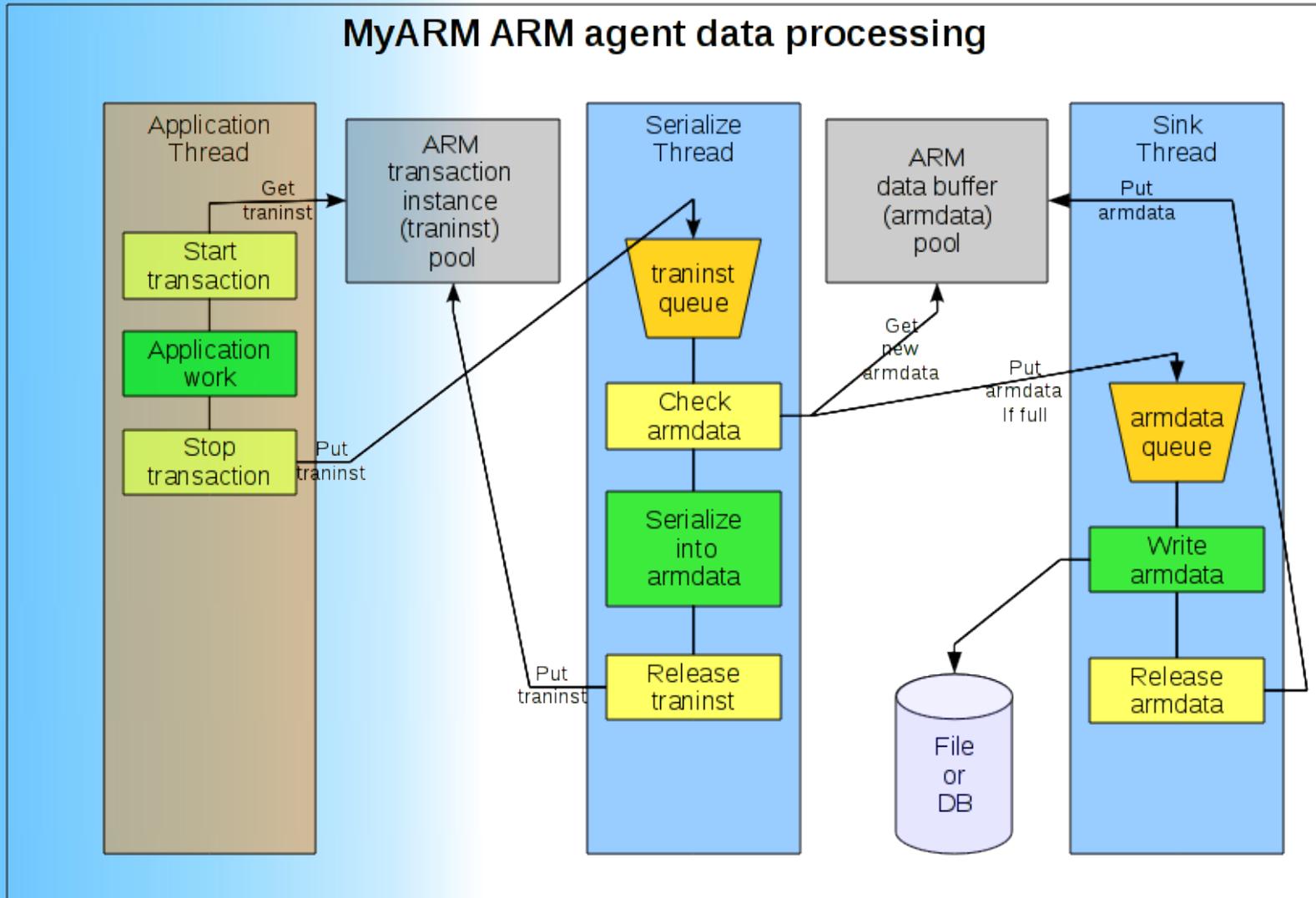


ARM APDEX



$$Apdex_T = \frac{\text{Satisfied count} + \left(\frac{\text{Tolerating count}}{2}\right)}{\text{Total count}}$$

MyARM Manager Korrelation



MyARM Manager Korrelation

MyARMManager::2, Selection: Tree

File Edit View Selections Database Windows Help

Tran DB: MyARM_myarm_info

Transaction filter: *

Transactions: 8 HasChildren Sort by: None Descending

Name	No	Identification	StartDate	StartTime	Duration	Status	Duration %	C: Artist	C: Album
httpd	101	Fetch-Discs	21.11.2009	18:09:36.737	165.595	GOOD	59.56 %	madonna	
HTTP	1678	Generate-Output	21.11.2009	18:09:36.904	1.649	GOOD	0.59 %		
PyCDDB	9	HTTP	22.11.2009	15:30:33.447	299.210	GOOD	100.00 %		
CDDB-Query	9	CDDB-Query	22.11.2009	15:30:33.586	39.214	GOOD	13.11 %		
DB-Connect	9	DB-Connect	22.11.2009	15:30:33.586	31.890	GOOD	10.66 %		
Fetch-Discs	9	Fetch-Discs	22.11.2009	15:30:33.618	4.742	GOOD	1.58 %	Queen	
Fetch-Tracks	157	Fetch-Tracks	22.11.2009	15:30:33.618	0.186	GOOD	0.06 %		A Day At The ...
Generate-Output	9	Fetch-Tracks	22.11.2009	15:30:33.618	0.186	GOOD	0.06 %		Queen II
		Fetch-Tracks	22.11.2009	15:30:33.618	0.195	GOOD	0.07 %		Star profile
		Fetch-Tracks	22.11.2009	15:30:33.619	0.180	GOOD	0.06 %		Innuendo
		Fetch-Tracks	22.11.2009	15:30:33.619	0.196	GOOD	0.07 %		A Day At The ...
		Fetch-Tracks	22.11.2009	15:30:33.619	0.219	GOOD	0.07 %		Live at Wembl...
		Fetch-Tracks	22.11.2009	15:30:33.619	0.188	GOOD	0.06 %		Jazz
		Fetch-Tracks	22.11.2009	15:30:33.619	0.177	GOOD	0.06 %		Innuendo (784)
		Fetch-Tracks	22.11.2009	15:30:33.621	0.213	GOOD	0.07 %		Greatest Hits P...
		Fetch-Tracks	22.11.2009	15:30:33.621	0.206	GOOD	0.07 %		Greatest Hits, ...
		Fetch-Tracks	22.11.2009	15:30:33.621	0.152	GOOD	0.05 %		Princes Of The...
		Fetch-Tracks	22.11.2009	15:30:33.621	0.189	GOOD	0.06 %		Greatest Hits III
		Fetch-Tracks	22.11.2009	15:30:33.621	0.199	GOOD	0.07 %		The Platinum ...
		Fetch-Tracks	22.11.2009	15:30:33.621	0.200	GOOD	0.07 %		A Kind Of Magic
		Fetch-Tracks	22.11.2009	15:30:33.621	0.180	GOOD	0.06 %		News Of The ...
		Fetch-Tracks	22.11.2009	15:30:33.621	0.173	GOOD	0.06 %		Hot Space
		Fetch-Tracks	22.11.2009	15:30:33.621	0.187	GOOD	0.06 %		The Game
		Fetch-Tracks	22.11.2009	15:30:33.622	0.174	GOOD	0.06 %		Made In Heaven
		Fetch-Tracks	22.11.2009	15:30:33.622	0.182	GOOD	0.06 %		Jazz
		Fetch-Tracks	22.11.2009	15:30:33.622	0.175	GOOD	0.06 %		Made In Heaven
		Generate-Output	22.11.2009	15:30:33.622	2.470	GOOD	0.83 %		
		HTTP	26.11.2009	10:28:35.124	673.918	GOOD	100.00 %		
		HTTP	30.11.2009	23:41:10.020	198.885	GOOD	100.00 %		
		HTTP	30.11.2009	23:42:35.914	385.094	GOOD	100.00 %		

Filters: App Tran Apply

Defined filters: default

Add Views: [Icons]

MySQL



Nonfunktionale Tests

- Nonfunktionale Tests, wenn vorhanden
 - Oft manuell durchgeführt
 - Ad hoc
 - Zeitintensiv
 - Unscharf formuliert

Motivation (III)

- Nonfunktionale Regressionstests nur mit hohem Aufwand machbar
- Wie kommt man zu Aussagen über Skalierbarkeit und Fehlertoleranz?
 - Bei der Erstellung der Architektur
 - Entwicklungsentscheidungen
 - Test
 - Betrieb

Motivation (IV)

- Informationen bzgl. Performance etc. werden meist aus Log-Dateien extrahiert
 - Unscharf (Granularität der Zeitstempel)
 - Langsam (IO-Systeme etc.)
 - Unflexibel (Skripte)
 - Geringe Auflösung (lediglich Start&Stop)
 - Wie korreliert man Ereignisse in anderen Systemen?