

SuDuelKu

Fault Tolerant Distributed Systems

	6		1		4		5	
		8	3		5	6		
2								1
8			4		7			6
		6				3		
7			9		1			4
5								2
		7	2		6	9		
	4		5		8		7	

Team 1

Saul Jaspán
Yudi Nagata
Lucia de Lascurain
Chris Nelson
Luis Rios

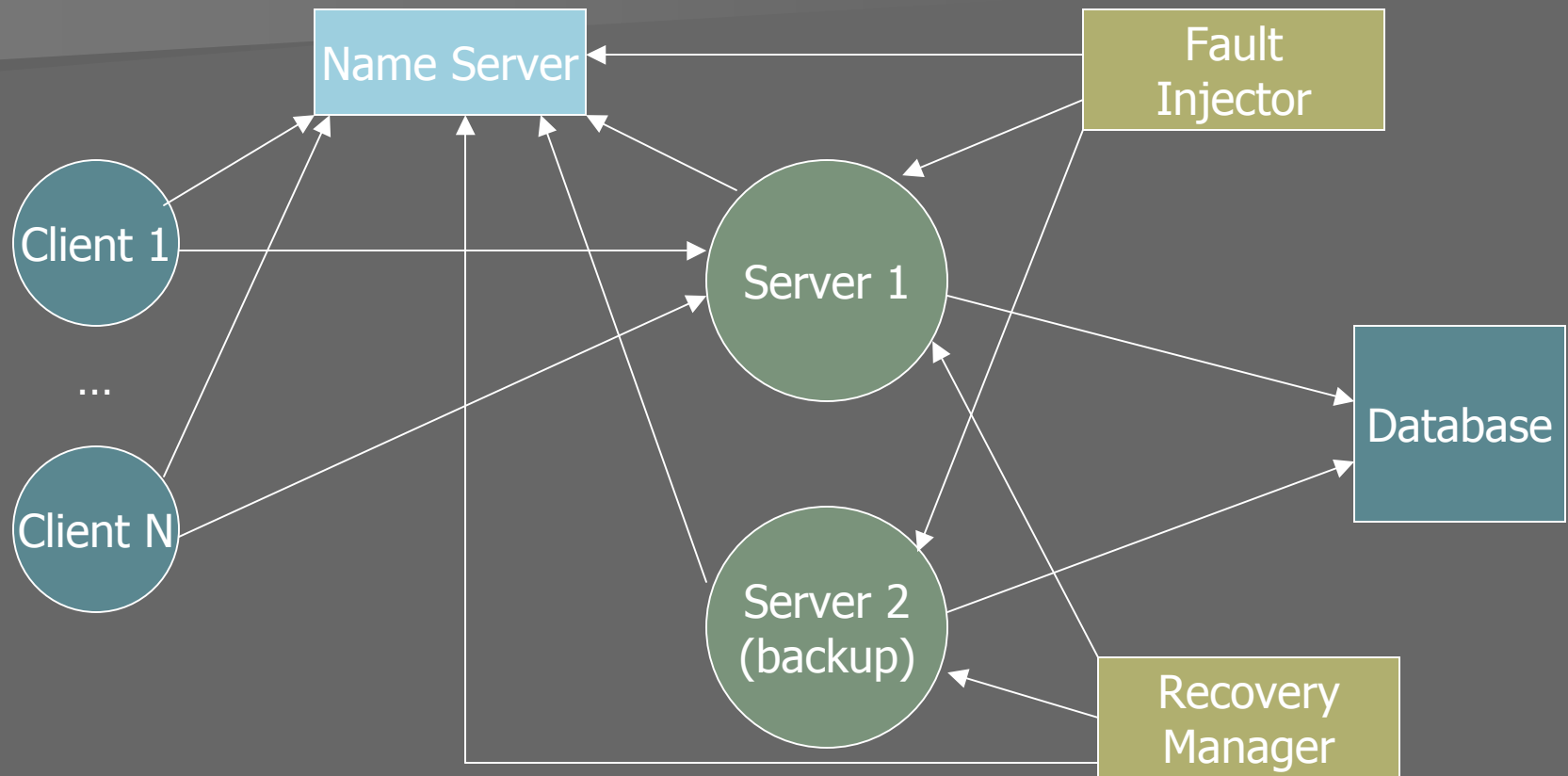
Agenda

- Application Configuration
- Fault Tolerant Architecture
- Questions and Assumptions
- Lessons learned
- What's next

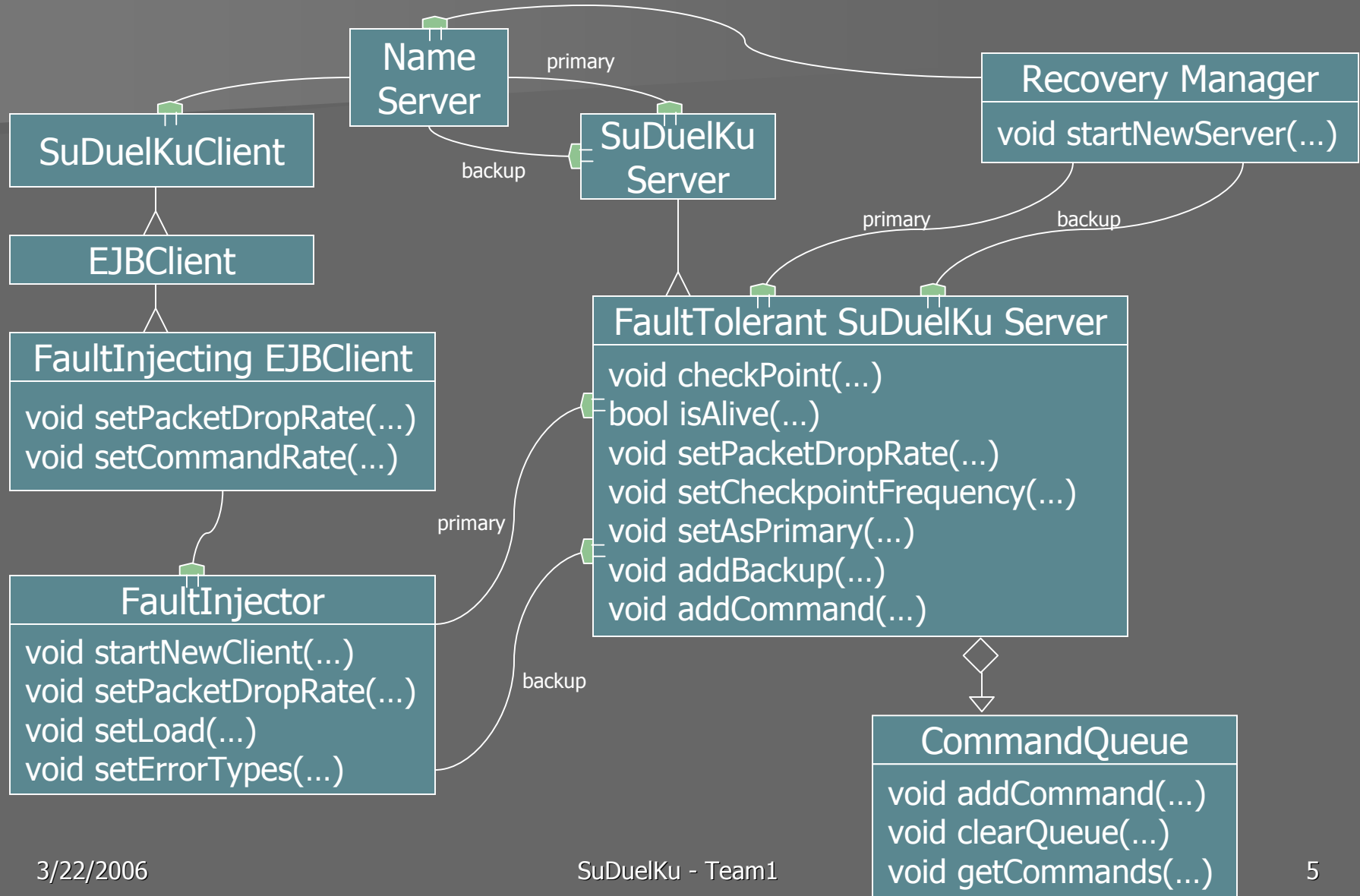
Application Configuration

- EJB
- Linux
- Java
- Using:
 - JBoss
 - MySQL
 - Lomboz
 - Eclipse

Fault Tolerant Architecture - 1



Fault Tolerant Architecture - 2



Questions and Assumptions

■ Questions:

- Are we really 100% stateless?
- Do we need checkpointing if we are stateless?
- Should we have a separate JNDI?

■ Assumptions:

- Our naming server will not be killed.
- Our message IDs will be committed to the database with the transactions atomically.

Lessons Learned

- EJB learning curve steep but eases development once summited
- Calling back from the server to the client is unnatural with middleware
- Fault Recovery is MUCH easier when the middle tier is stateless

What's next

- Develop fault tolerant system.
- Develop fault injector.
- Obtain system metrics.

Questions

