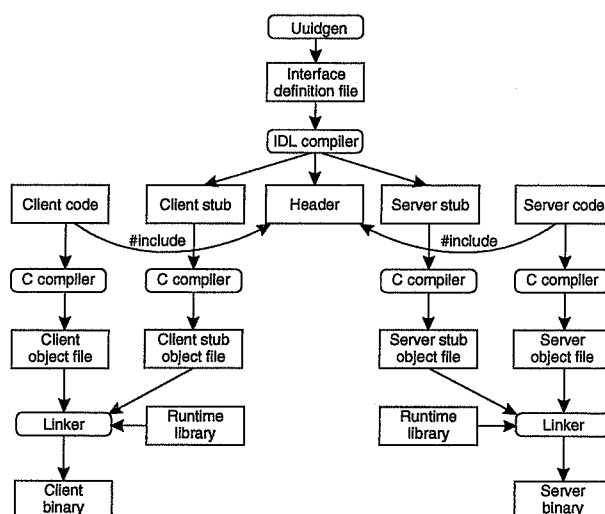


- 1a Sommige gedistribueerde systemen kunnen eenvoudigweg niet schalen, ongeacht de toegepaste techniek. Geef een voorbeeld van een dergelijk systeem en leg uit waarom schaalbaarheid (bijna) onmogelijk is. 5pt
- 1b Leg uit hoe code migratie zoals gedaan wordt bij b.v. Java applets in het geval van Web services kan helpen om schaalbaarheid te verbeteren. Welk schaalbaarheidsprobleem wordt hiermee aangepakt? 5pt
- 2a Leg het verschil uit tussen een *process virtual machine* en een *virtual machine monitor*. 5pt
- 2b Leg uit hoe virtuele machine technologie toegepast wordt in PlanetLab, en hoe die technologie helpt om meerdere applicaties tegelijk te laten draaien. 5pt
- 2c Virtuele machines helpen om applicaties te migreren tussen verschillende servers. Waarom is dit gemakkelijker dan algemene oplossingen voor sterke mobiliteit (Eng.: *strong mobility*)? 5pt
- 3a Leg uit welke functies geïmplementeerd zijn in de runtime bibliotheek van een RPC systeem. Zie ook onderstaande figuur. 5pt



- 3b Om een RPC uit te voeren moet een client contact leggen met een server. Hoe wordt zo'n contactpunt gevonden en hoe ziet het er concreet uit? 5pt
- 3c RPC systemen kunnen geen lokale referenties ondersteunen (zoals pointers), omdat deze naar objecten in het lokale geheugen refereren. In plaats daarvan moeten globale object referenties gebruikt worden, indien mogelijk. Schets de implementatie van zo'n referentie. 5pt
- 4a Leg het principe uit van een epidemisch protocol. 5pt
- 4b Schets hoe knopen in een gedistribueerd systeem elk de grootte van het systeem (i.e., het totaal aantal knopen) kunnen berekenen d.m.v. een epidemisch protocol. 5pt
- 4c Wat is het probleem met het verwijderen van data in een epidemisch systeem en hoe kan dit probleem opgelost worden? 5pt
- 5a Waarom is het onderstaande opslagsysteem niet sequentieel consistent? Is het causaal consistent? Zorg dat je je antwoorden uitlegt. 5pt

P1:	W(x)a		
P2:	W(x)b		
P3:	R(x)b	R(x)a	
P4:		R(x)a	R(x)b

- 5b Beschouw een systeem dat *read-your-writes* consistentie combineert met *writes-follow-reads*. Is dit systeem ook sequentieel consistent? Leg je antwoord uit. 5pt
- 5c Consistentie kan ook geformuleerd worden in termen van numerieke afwijkingen. Geef een voorbeeld van zo'n vorm van consistentie en schets hoe het ook opgelegd kan worden. 5pt
- 6a Wat is een k fout-tolerante groep en hoe hangt k af van *failure semantics*? 5pt
- 6b Laat zien dat drie communicerende processen, waarvan er één niet goed werkt, niet voldoende is om overeenstemming te laten krijgen tussen de overgebleven twee als we met Byzantijnse fouten te maken hebben. 5pt
- 7c Leg uit wat een *flash crowd* is, en waarom het zo moeilijk is om een algemene voorspeller te ontwikkelen. 5pt
- 7b Wat is de beste oplossing om flash crowds op te vangen. Leg kort en bondg uit hoe je oplossing werkt. 5pt

Cijferbepaling: Het cijfer wordt bepaald door de punten die per onderdeel behaald zijn, bij elkaar op te tellen (totaal maximaal 90 punten), en vervolgens daar 10 extra punten bij te tellen. Er zijn dus totaal 100 punten te behalen.