

Übung zur Vorlesung „Verteilt-kooperative Informationsverarbeitung“ - SS 2007

Blatt 4

Ausgabe 21.05. - Abgabe 11.06.

Aufgabe 4.1 – 10 Punkte

- Warum genügt der MAF-Spezifikation bei Stellen ein unspezifischer String, während für Agentensysteme ein Location-Objekt gewählt wird? (Auch Location ist ein String, aber es wird eine gewisse Struktur vorausgesetzt.)
- Warum führte man den MAFFinder ein, obwohl es bereits CosNaming gibt?
- Nehmen Sie an, dass ein Rechner nur temporär mit dem Internet verbunden wird und bei jeder Einwahl eine neue IP-Adresse erhält. Dieser Rechner betreibe ein MAF-Agentensystem. Beschreiben Sie das sich ergebende Problem sowie eine mögliche Abhilfe.

Aufgabe 4.2 – 10 Punkte

Implementieren Sie mittels Pseudocode die Migration zwischen zwei MAF-konformen Stellen. Gehen Sie davon aus, dass die Systeme eine CORBA-Java-Abbildung verwenden, in der Sie die entfernten Methoden auf einem Stub-Objekt so aufrufen können, wie sie in der IDL-Datei beschrieben sind.

Hilfestellung: Überlegen Sie sich, welche Schritte bei der Migration stattfinden müssen. Suchen Sie geeignete Methoden aus MAFAgentSystem. Implementieren Sie eine migrate-Methode in der lokalen Stelle, welche die entsprechenden MAF-Methoden auf dem Stub aufruft sowie eine entsprechende Methode in der entfernten Stelle. Wählen Sie eine Möglichkeit des Klassentransfers. Versuchen Sie, Ihren Pseudocode so „realistisch“ wie möglich zu gestalten, d.h. machen Sie sich u.a. auch Gedanken, woher Sie die notwendigen Referenzen bekommen.

Aufgabe 4.3 – 15 Punkte

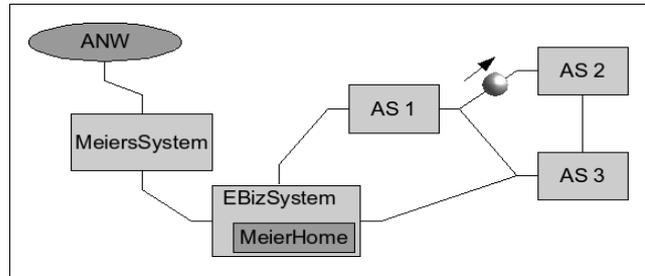
Wir spielen nun ein Szenario durch, in dem mobile Agenten in einem MAF-konformen System zum Einsatz kommen sollen. Bitte geben Sie die beteiligten Objekte, Methoden und Datenstrukturen an, etwa in Form eines Pseudocodes mit Kommentaren (ähnlich wie in Aufgabe 2). Die unterstrichenen Punkte im Text deuten jene Momente an, die in dieser Weise zu beschreiben sind; Sie können darüber hinaus weitere Vorgänge formal beschreiben. Wichtig ist jedoch, dass Sie die in der MAF-Spezifikation definieren Methodennamen und Strukturen angeben. Beispiel:

```
MeierHome: {  
    // gerade gestartet  
    MAFFinder finder = eBizSystem.get_MAFFinder();  
    finder.register_place("MeierHome", eBizLocation); ...  
}
```

Herr Meier möchte eine bestimmte Ware kaufen und schickt einen Agenten los, um sich ein Angebot einzuholen. Er startet seine Anwendung (ANW). Ein lokales Agentensystem „MeiersSystem“ ist bereits am Laufen, und ANW nutzt dieses System, um weitere Agentensysteme zu finden. Genauer gesagt benötigen wir die Stelle „MeierHome“ auf einem der erreichbaren Agentensysteme (deren Namen nicht

bekannt ist). Diese Stelle wurde zuvor gestartet und hat sich registriert. (siehe oben)

Ein Agent wird auf „MeierHome“ gestartet. Ihm werden zum Start von ANW Parameter übergeben. Er registriert sich. Der Agent soll nun herumreisen. ANW hat dem Agenten eine Route mitgegeben.



Der Agent reist zum nächsten Ort. Er ruft eine migrate-Methode auf seiner Stelle auf und löst damit die Migration aus. Der Zielort verfügt jedoch nicht über alle Klassen und bittet das abschickende System, die Klassen nachzuschicken.

Es ist ein wenig Zeit vergangen, und Herr Meier möchte wissen, ob sein Agent noch lebt oder irgendwo abgestürzt ist. Es ist aber nicht bekannt, auf welcher Stelle der Agent weilt, lediglich die Menge der Agentensysteme ist bekannt. ANW geht systematisch vor und testet diese Agentensysteme der Reihe nach.

Der Agent ist gefunden und läuft noch. ANW kontaktiert den Agenten, der seine Daten applikationsspezifisch liefert. (Wir gehen davon aus, dass der Agent die speziellen Methode „getData“ und „setData“ besitzt, um Daten zu liefern oder entgegenzunehmen.) Diese Daten verraten, dass er ein Agentensystem ausließ. Das ausgelassene System nimmt offenbar zurzeit keine Agenten an, ist aber ansprechbar. Er lässt den Agenten über ANW anhalten.

ANW bittet den Agenten, eine geeignete Stelle in der Nähe (des ausgelassenen Systems) aufzusuchen. ANW lässt den Agenten weiterlaufen. Der Agent hat alle Angebote eingesammelt und kehrt zurück zu „MeierHome“. Er liefert Herrn Meier die Angebote, und ANW stoppt den Agenten endgültig.

Aufgabe 4.4 – 12 Punkte

Nun versuchen wir, das Szenario aus Aufgabe 3 in AMETAS zu entwerfen. Dabei sollte wie folgt vorgegangen werden:

- Analysieren Sie, welche Agenten beteiligt sind. Welche Rolle spielt der Anwender?
- Analysieren Sie die Anwendungsfälle, die im System auftreten.

Anwendungsfälle zeigen auf, welche Operationen vom Anwender ausgelöst werden. Dies dient dazu, sich einen Überblick zu verschaffen, wie der Funktionsumfang der Komponenten zu gestalten ist.

- Welche Interaktionen treten auf? Stellen Sie die Nachrichten zusammen sowie die zugehörigen Protokolle (Nachrichtenabfolgen). Sie können dies mit Hybridtypbeschreibungen lösen.

Nachdem nun der systemunabhängige Teil des Entwurfs geleistet ist, wenden Sie diese Erkenntnisse dazu an, das System in AMETAS zu erstellen.

- Entwerfen Sie ein System von AMETAS-Stellen, welches das obige Szenario realistisch umsetzt.
- Welche Stellennutzer kommen für welche Komponenten des Systems zum Einsatz? (Achten Sie darauf, dass Sie Nachrichten nur lokal senden können.)
- Entwerfen Sie eine einfache Benutzerschnittstelle (d.h. skizzieren Sie die Oberfläche inklusive Menü/Knöpfe/Textfelder sowie Verbindungen zu Komponenten).

Hinweis. Diese Aufgabe erwartet keine tatsächliche Implementierung, sondern ist konzeptioneller Natur. Eine Implementierung kann in Aufgabe 4.5 geleistet werden.

Aufgabe 4.5 – 10 Zusatzpunkte (freiwillig)

Implementieren Sie Ihren Entwurf aus Aufgabe 4.4. für AMETAS.

Einige der Funktionen aus Aufgabe 3 sind nicht unmittelbar in AMETAS implementierbar. Es gibt beispielsweise keine Suchfunktion für Agenten, die irgendwo im Netz wandern. Dennoch ist das obige Beispiel ohne Einschränkungen in Bezug auf alle Anwendungsfälle realisierbar. Nutzen Sie die Möglichkeiten der Schnittstelle `AMETAS.place.AMETASPlaceUserDriverIf`. Bitte vermeiden Sie die „deprecated“-Methoden, insbesondere `depositRemoteMessage`. Finden Sie einen Weg, Ihrem entfernt laufenden Agenten eine Nachricht mitzuteilen.

User's Guide: <http://www.ametas.de/en/docs/UsersGuide/>; API: <http://www.ametas.de/docs/apipub/>