

Übung Netzwerke und Agenten:

Projekt-Themen

Projekt 1: DAML+OIL

Für WeLearn Foren-Beiträge (ähnlich zu Diskussionsgruppen) bzw. Listen davon sowie für ganze Foren soll eine Definition in DAML+OIL erstellt werden. Der Text enthält einige wenige Formatierungsanweisungen (wieHTML): `<bold>`, `<italic>`, ... Weiters sollen Beispiels-Dateien dazu erstellt werden, welche die verschiedenen Möglichkeiten aufzeigen und zum Testen dienen können.

In Bezug auf das Agentensystem soll ein Agent erstellt werden, der ein Benutzerinterface anbietet, mit dem solche Beiträge (ohne Kenntnis der Darstellung!) erstellt werden können. Diese werden anschließend nach der erstellten Spezifikation codiert und über eine Nachricht an einen anderen Agenten gesendet. Dieser liest die Nachricht ein, decodiert sie, und gibt die einzelnen Teile getrennt in eine Textdatei aus. Beide Agenten sollen auch in die „andere Richtung“ funktionieren: Auf einen Mausklick hin wird eine Datei mit Beiträgen eingelesen, codiert und an den GUI-Agenten geschickt, welcher diese speichert und auf Anfrage dem Benutzer anzeigt.

Diese Funktionalität wird für die Erweiterung des WeLearn Systems benötigt, daher ist dies von großer praktischer Bedeutung. Das Projekt kann ev. auch als 10-stündiges Praktikum und Diplomarbeit weitergeführt werden.

Projekt 2: Registrar

Es ist ein Agent zu erstellen, der als Registrierungsdienst für Agenten dient. Agenten melden sich dort mit ihren Fähigkeiten an (Beschreibung in DAML+OIL; diese ist ebenfalls in sehr einfacher Form zu erstellen). Agenten fragen dort ebenso nach anderen Agenten mit bestimmten Fähigkeiten (in gleicher Weise spezifiziert). Die Spezifikation muß frei erweiterbar sein und nur einige Beispiele anbieten.

Zwei weitere Agenten sind zu programmieren: Einer, der sich mit seinen Fähigkeiten (Auswahl über Benutzerinterface) an- und abmelden kann, sowie ein weiterer Agent, der entsprechende (ebenso GUI-konfigurierte) Abfragen erzeugen kann. Als Ergebnis soll zwischen den gefundenen Agenten eine triviale Kommunikation stattfinden, um die Korrektheit zu überprüfen.

Die Auswahl der Agenten für eine gegebene Beschreibung im Registrar selbst soll modular sein, sodaß sie einfach ersetzt werden kann. Zu implementieren ist nur das Erstellen einer Liste mit allen Agenten, die exakt passende Fähigkeiten aufweisen.

Projekt 3: Weblinks prüfen

Es ist ein Agent zu erstellen, der Webseiten nach ungültigen Links durchsucht und sie diesfalls bearbeitet/korrigiert. Hierzu ist sowohl der URL zu den Webseiten (für die Auflösung der Links zu anderen Seiten) als auch der lokale Pfad (um die Dateien ändern zu können) anzugeben. Weiters ist eine Beschränkung der Suchtiefe (wie tief in Subdirectories die Webseiten noch geprüft werden) einzubauen.

Der Agent lädt die Webseiten vom Laufwerk, durchsucht sie nach Links (in allen Varianten), erstellt für diese vollständige URLs und prüft, ob diese Seite geladen werden kann. Falls nicht, wird die Originalseite so verändert, daß der Link auf eine spezielle Seite zeigt, die eine Liste aller nicht mehr gültigen Links enthält. Auf dieser Liste ist der Original-Link dann einzutragen. Alle Vorgänge müssen genau protokolliert werden, da ev. eine Erweiterung erfolgen soll, sodaß nicht sofort sondern erst bei einem späteren wieder erfolglosen Versuch diese Änderung vorgenommen wird.

Kann die Seite zwar geladen werden, erfolgt jedoch ein automatisches Redirect, so soll der Original-Link durch den Redirect-Link ersetzt werden. Dies soll als abschaltbare Option integriert werden.

Projekt 4: Dublin-Core Digester Erweiterung

Die Übung 2 soll so erweitert werden, daß auf den Webseiten auch alle Links identifiziert und nachverfolgt werden (d. h. eine Art Spider-Funktionalität). Bis zu einer (in der Nachricht) vorgegebenen Tiefe sollen alle Dokumente auf diese Art ausgewertet werden. Auch für Bilder oder sonstige Ressourcen soll versucht werden, zumindest minimale Informationen bereitzustellen. Diese Auswertung einer Datei soll als eigenes unabhängiges Modul implementiert werden.

Zusätzlich sollen die Ergebnisse auch in eine Text-Datei geschrieben und von dort wieder eingelesen werden. Hierzu ist auch ein weiterer Agent zu schreiben, der nur diese Datei ausliest, nach bestimmten Informationen darin suchen kann, und diese dann für den Benutzer darstellt.