

```

ance: title: simple body mass index;; mlname: BMI;; arden: Version 2.8;; version: 1.00;; institution: Medexter Healthcare;; author:
specialist: ;; date: 2012-02-15;; validation: testing;; library: purpose: body mass index;; explanation: input: compound list with: (n
in m, (number) weight in kg, (time) birth date. output: the classification wrt. WHO;; keywords: BMI, body mass index;; citations: ;;
n.wikipedia.org/wiki/Body_mass_index;; knowledge: type: data driven;; data: (size, weight, birth) := argument;; priority: ;; evoke: ;;
be weight / (size ** 2); // BMI age := currenttime - birth; // AGE if the age is less than 19 years then classification := null;
is less than 18.5 then classification := localized 'under'; elseif the bmi is less than 25 then classification := null; // BMI
ise let the classification be localized 'over'; endif; bmi := bmi formatted with localized 'msg'; conclude classification is
logic: let bmi be weight / (size ** 2); // BMI
let bmi be weight / (size ** 2); // BMI
age := currenttime - birth; // AGE
if the age is less than 19 years then
classification := null;
elseif the bmi is less than 18.5 then
classification := localized 'under';
elseif the bmi is less than 25 then
classification := null; // BMI normal range
elseif the bmi is less than 25 then
classification := null; // BMI normal range
else
let the classification be localized 'over';
endif;

```

Klaus-Peter Adlassnig, Karsten Fehre

Arden-Syntax zur Repräsentation und Verarbeitung medizinischen Wissens

Einleitung

Ein grundlegendes Ziel der Medizinischen Informatik ist es, einen wesentlichen Beitrag zu umfassendem Qualitätsmanagement im Rahmen der Patientenbehandlung zu leisten. Dabei kommen die unterschiedlichsten Methoden der computergestützten Qualitätssicherung und -verbesserung, der Fehlervermeidung sowie der Kostenoptimierung zum Einsatz.

Einen wichtigen Ansatz zur Erfüllung dieses Ziels bilden Systeme zur klinischen Entscheidungsunterstützung (Englisch: clinical decision support systems). Diese können in vielen Bereichen der medizinischen Behandlung des Patienten – im Krankenhaus, im Ambulatorium oder beim niedergelassenen Arzt – eingesetzt werden. Auf der Grundlage von aktuell vorliegenden Patientendaten und gespeichertem medizinischen Wissen sind diese Systeme in der Lage, dem behandelnden medizinischen Personal Unterstützung in Diagnostik, der Auswahl einer optimalen Therapie, der korrekten Einschätzung der Prognose sowie des effektiven Patientenmanagements bereitzustellen. Durch die

Anwendung formaler Methoden der Wissensrepräsentation und des Wissenserwerbs ist es möglich, das dazu benötigte medizinische Wissen in hoch strukturierter Form in medizinischen Wissensbasen anzubieten. So können diese Systeme unter Nutzung der jeweiligen Wissensbasis einerseits einzelne Aspekte der Patientendaten interpretieren, andererseits aber auch eine umfassende, ganzheitliche Betrachtung der vorliegenden Patientendaten liefern. Diese aufbereiteten Informationen können anschließend vom medizinischen Personal zur personalisierten Entscheidungsunterstützung genutzt werden. In [1] wurde gezeigt, dass solche Systeme zur klinischen Entscheidungsunterstützung die Qualität der Patientenbehandlung deutlich verbessern können.

Eine Grundvoraussetzung dafür ist jedoch, dass das hierzu notwendige medizinische Wissen in einer Form vorliegt, die für den Computer verarbeitbar ist. Zu diesem Zweck wurde für medizinische Anwendungen die „Arden Syntax for Medical Logic Systems“, eine von Health Level Seven International (HL7) [2] betreute Standardsprache zur Repräsentation und Verarbeitung von medizinischem Wissen, entwickelt.

Arden-Syntax

Während einer Klausurtagung im Arden Homestead Retreat in Orange County, NY, USA, im Jahr 1989 entstand die Erstversion der Arden-Syntax. Sie wurde im Jahr 1992 von der American Society for Testing and Materials (ASTM) genehmigt und als Standard E-1460-92 veröffentlicht. Im Jahr 1999 wurde die Sprache zur Pflege und Weiterentwicklung von der HL7-Organisation übernommen, die nach einigen Neuerungen die Version 2.0 publizierte. Sie wird seitdem von der HL7 Arden Syntax Working Group betreut [3]. Die jeweils entstehenden Neuversionen werden regelmäßig auch als Standard des American National Standards Institute (ANSI) anerkannt.

Die Arden-Syntax wurde zur formalisierten Darstellung von medizinischem Wissen und seiner Verarbeitung und damit speziell für den Aufbau und den praktischen Einsatz von Entscheidungsunterstützung in medizinischen Einrichtungen entwickelt. Ein wesentlicher Punkt bei der Formulierung dieser Sprache war auch der Gedanke, fertige Wissensmodule zwischen verschiedenen medizinischen Einrichtungen austauschen zu können. Außerdem enthält diese Sprache Funktionen und Konzepte, die den Einsatz auf dem Gebiet der Medizin erleichtern. Beispielsweise existieren spezielle, für medizinische Belange äußerst sinnvolle Operatoren auf Listen wie z. B. ein Operator zum Sortieren von Listen – wahlweise nach den Werten der Listenelemente oder deren so genannte Primary Time (ein Zeitstempel) – oder der „where“-Operator, der ein direktes „Select“ auf der ihm übergebenen Liste ausführt. Der Umstand, dass für jeden Wert bzw. Parameter eine Primary Time angegeben werden kann, ermöglicht es z. B., dass Symptom-, Krankheits- und Therapieverläufe programmtechnisch einfacher nachzubilden sind.

Um die Formalisierung bestehenden Wissens zu erleichtern, legt die Arden-Syntax großen Wert auf die Lesbarkeit der Programmzeilen für den Menschen. Die syntaktischen Konstrukte in der Arden-Syntax sind dementsprechend verständlich gehalten. Damit soll eine vereinfachte Verifikation des formalisierten klinischen Wissens durch Experten des entsprechenden medizinischen Wissensgebietes ermöglicht werden.

Medical Logic Modules

Um medizinisches Wissen in der Arden-Syntax zu repräsentieren, werden logisch zusammengehörige medizinische Entscheidungsschritte zu Regeln zusammengefasst, die dann in Medical Logic Modules (MLMs) festgehalten werden. Die Verbindung mehrerer solcher MLMs zu einer Wissensbasis (wir nennen solche Bündel auch Medical Knowledge Package (MKP) [4]) dient üblicherweise der computergestützten Repräsentation eines bestimmten abgeschlossenen Teilgebietes des medizinischen Wissens. Ein MLM selbst enthält nicht nur rein medizinisches Wissen, sondern auch Metainformationen darüber, so z. B. administrative und bibliographische Angaben. Zu diesem Zweck sind die MLMs in mehrere Kategorien unterteilt, die jeweils zum Festhalten

bestimmter Informationen entworfen wurden. Die folgenden vier Kategorien können, bzw. müssen in einem MLM angege- ben werden:

- ▶ **Maintenance:** Hier werden administrative Details zum aktuellen MLM festgehalten, wie Name des MLMs, Institution, die das MLM benutzt oder erstellt hat, Version und Autor des MLMs und benutzte Version der Arden-Syntax.
- ▶ **Library:** Diese Kategorie enthält beschreibende Informationen über den Inhalt des MLMs, z. B. eine textliche Erläuterung der formalisierten Regel, Quellen und Links zu Hintergrundinformationen sowie das beabsichtigte, entscheidungsunterstützende Ziel des MLMs.
- ▶ **Knowledge:** Die Knowledge-Kategorie enthält die eigent- liche Logik des MLMs und die auszuführenden Aktionen bei erfolgreicher Abarbeitung der Regel.
- ▶ **Resources:** Hier können Texte länderspezifisch in verschiedenen Sprachen angegeben werden, die dann innerhalb der MLM-Logik verwendet werden.

Die einzelnen Kategorien wiederum sind in so genannte Slots unterteilt, die jeweils zur Aufnahme weiterer detaillierter Informationen bestimmt sind.

Datenzugriff und „Curly-Brace“-Ausdrücke

Der Anstoß zur Ausführung eines MLMs kann aufgrund bestimmter Steuerungs-, Daten- oder Zeitereignisse erfolgen.

So kann ein MLM entweder von einer externen Quelle wie z. B. über eine Webservice-Schnittstelle direkt aufgerufen, von einem anderen MLM konsultiert oder durch ein festgelegtes Event getriggert werden. Hierbei können vom MLM zu verarbeitende Daten direkt an das MLM übergeben werden. Alternativ kann ein MLM über so genannte „Curly-Brace“-Ausdrücke (sie enthalten gewöhnlich institutionsspezifische Datenbankzugriffe) auch auf externe Datenquellen zugreifen. Diese Ausdrücke werden an das umgebende System weitergeleitet, das diese interpretiert und die angeforderten Daten an das MLM zurückliefert. Ergebnisse der Ausführung eines MLMs können wiederum direkt an den Aufrufer zurückgesandt oder mittels „Curly-Brace“-Ausdrücken in eine externe Datenenke geschrieben werden.

Das so genannte „Curly-Brace“-Problem beruht nun auf der Tatsache, dass keine festgeschriebene Syntax für die „Curly-Brace“-Ausdrücke existiert. Bei der Erstellung von MLMs werden die nötigen „Curly-Brace“-Ausdrücke meist in Hinblick auf die Zielumgebung entworfen, was zur Folge hat, dass ein inter- institutioneller Austausch von Einzel-MLMs oder von ganzen Wissensbasen erheblich erschwert wird. Bestehende MLMs müssen unter Umständen an die neue Umgebung angepasst werden, um den vorliegenden Datenquellen und deren Struktur zu entsprechen. Die mit der Weiterentwicklung der Arden-Syntax beauftragte Arden Syntax Working Group der HL7-Organisation arbeitet seit einiger Zeit an einer Lösung dieses Problems [5, 6].

Das folgende Beispiel-MLM berechnet den Body-Mass-Index (BMI) eines Patienten aus den übergebenen Parametern und zeigt den grundsätzlichen Aufbau eines MLMs (Tabelle 1).

```

maintenance:
  title: simple body mass index;;
  mllname: BMI;;
  arden: Version 2.8;;
  version: 1.00;;
  institution: Medexter Healthcare;;
  author: Karsten Fehre;;
  specialist: ;;
  date: 2012-02-15;;
  validation: testing;;
library:
  purpose: body mass index;;
  explanation: input: compound list with: (number) size
    in m, (number) weight in kg, (time) birth date.
    output: the classification wrt. WHO;;
  keywords: BMI, body mass index;;
  citations: ;;
  links: http://en.wikipedia.org/wiki/Body_mass_index;;
knowledge:
  type: data driven;;
  data: (size, weight, birth) := argument;;
  priority: ;;
  evoke: ;;
  logic:
    let bmi be weight / (size ** 2); // BMI
    age := currenttime - birth; // AGE
    if the age is less than 19 years then
      classification := null;
    elseif the bmi is less than 18.5 then
      classification := localized 'under';
    elseif the bmi is less than 25 then
      classification := null; // BMI normal range
    else
      let the classification be localized 'over';
    endif;
    bmi := bmi formatted with localized 'msg';
    conclude classification is present ;
  ;;
  action:
    write bmi || classification || ".";
    return classification;
  ;;
  urgency: ;;
resources:
  default: de;;
  language: en
  'msg' : "The patient's BMI %.1f is not in the
    normal range and is classified as being";
  'under' : "underweight.";
  'over' : "overweight."
  ;;
  language: de
  'msg' : "Der BMI %.1f ist nicht im Normalbereich
    und wird klassifiziert als ";
  'under' : "Untergewicht.";
  'over' : "Übergewicht."
  ;;
end:
  
```

Tabelle 1: Beispiel-MLM zur Berechnung des Body-Mass-Index (BMI)

Entwicklungsschritte

Wie oben erwähnt, wird die Arden-Syntax stetig überarbeitet, weiterentwickelt und verbessert (Tabelle 2).

Eine der wichtigsten vergangenen Weiterentwicklungen an der Arden-Syntax war die Einführung von Objekten, die es erlauben, komplexe Datenstrukturen zu modellieren, zu bearbeiten und auszutauschen. Außerdem erleichtert die Einführung von verschiedenen sprachigen Texten durch Hinzufügen der „Resources“-Kategorie und des „localized“-Operators den Austausch von MLMs auch über Landesgrenzen hinaus. Hierzu werden für

Version (Jahr)	Änderung
2.1 (2002)	neue String-Operatoren; Keyword „currenttime“, um auf die aktuelle Systemzeit zugreifen zu können
2.5 (2005)	Erstellen und Bearbeiten von Objekten; XML-Schema zur Repräsentation von MLMs als XML-Dokumente
2.6 (2007)	Erweiterung der möglichen Kodierungen um UNICODE; Resources-Kategorie zur Definition von Texten in unterschiedlichen Sprachen; Datentypen Time-Of-Day und Day-Of-Week; neuer Operator „localized“, um auf Texte in unterschiedlichen Sprachen zugreifen zu können
2.7 (2008)	Erweiterung des Zuweisungsoperators; Erweiterung des Operators „new“, um die Instantiierung von Objekten flexibler und leichter lesbar zu gestalten
2.8 (2012)	neue Operatoren zur Manipulation von Listen wie „remove ... from ...“ und „add ... to ... [at] ...“, um das Entfernen und Hinzufügen von Listenelementen zu vereinfachen; Operatoren zur Manipulation von Teilen eines Datums; Switch-Statements, um die Darstellung einer Mehrfach-Verzweigung zu vereinfachen; Keyword „break“ zur Unterbrechung von Schleifen

Tabelle 2: Wichtige Entwicklungsschritte der Arden-Syntax in den letzten Jahren

jede gewünschte Zielsprache Texte bzw. Meldungen hinterlegt, die für Statusmeldungen oder Rückgabewerte genutzt werden können. Zum Zeitpunkt der Ausführung meldet das System, welches das MLM umgibt, die aktuelle Systemsprache an das MLM. Dadurch kann der der jeweiligen Sprache zugeordnete Text identifiziert und verwendet werden.

Implementierung der Arden-Syntax

Die Struktur der Arden-Syntax erfordert, dass ein MLM zum Zeitpunkt seiner Ausführung in der Lage sein muss, Informationen mit dem umgebenden System auszutauschen. Zusätzlich bedarf es einer Applikation, der alle auf dem System verfügbaren MLMs bekannt sind, sodass diese bei Eintreten von Trigger-events die betroffenen MLMs starten kann. Aus diesem Grund können MLMs nicht direkt in ausführbare Programme übersetzt werden. Ein MLM muss also entweder direkt zur Laufzeit interpretiert oder in eine Zwischensprache übersetzt werden.

Die Übersetzung in eine Zwischensprache hat den Vorteil, dass zur Laufzeit verfügbare MLMs schon im vorausgegangenen Compiler-Schritt syntaktisch verifiziert wurden und dass die für die Ausführung notwendige Information (z. B. auf welche Trigger-events ein MLM reagieren soll) schnell verfügbar ist.

Eine virtuelle Maschine, auf der die in der Zwischensprache vorliegenden MLMs ausgeführt werden, muss in der Lage sein, Anfragen eines MLMs zu bedienen. Dies können Aufrufe anderer

MLMs, Anfragen an eine Datenquelle oder Schreibzugriffe sein. Ebenso muss eine solche virtuelle Maschine alle für sie verfügbaren MLMs kennen und wissen, welche der MLMs auf ein spezifisches Event reagieren sollen. Die virtuelle Maschine muss zusätzlich in der Lage sein, die in der Arden-Syntax vorgesehene Definition von periodischen Aufrufen von MLMs verarbeiten zu können.

Modularer Aufbau und service-orientierte Architektur

Unterschiedliche Institutionen benutzen unterschiedliche Systeme für die Speicherung, Verwaltung und den Zugriff auf Daten. MLMs bzw. die MLM-ausführenden Systeme, die mit den Daten einer bestimmten Institution arbeiten sollen, müssen stark an das jeweilige vorhandene, spezifische System angepasst werden. Die Einführung einer modularen Architektur für die virtuelle Maschine bietet nun den Vorteil, dass nur die Datenzugriffskomponente an die Systemumgebung in einer neuen Institution angepasst werden muss.

Um die Interoperabilität und Integrationsfähigkeit eines auf Arden-Syntax basierenden, entscheidungsunterstützenden Systems zu erhöhen, müssen zusätzliche Schnittstellen geschaffen werden. Die Erweiterung der oben beschriebenen virtuellen Maschine auf eine service-orientierte Architektur (SOA) [7] bietet zusätzliche Vorteile:

- ▶ Fernzugriff auf Wissensbasen über ein internes Netz oder das weltweite Internet;
- ▶ das Service kann von jeder Applikation konsumiert oder leicht in bestehende Applikationen eingebunden werden;
- ▶ bessere Skalierbarkeit und vereinfachte Wartung der einzelnen Software-Komponenten, z. B. Web-Frontend, Service-Schnittstelle und konsumierende Kundenapplikation.

Eine solche Erweiterung ermöglicht eine Vielzahl von Einsatzmöglichkeiten. So ist ein zentrales Repository an Wissensbasen denkbar, welches über das Internet für jeden zugänglich ist und von unterschiedlichen Applikationen oder Institutionen genutzt werden kann. Zusätzlich ist die Portierung auf beliebige Plattformen (z. B. Web, Smart-Phones und Tablets) möglich.

Datenzugriff

Es besteht die Möglichkeit, dass die zu analysierenden, interpretierenden bzw. zu verarbeitenden Daten den MLMs direkt beim Aufruf übergeben werden, oder aber, dass sie über „Curly-Brace“-Ausdrücke aus einer Datenquelle gelesen werden. Zentrale und institutionsübergreifende Repositories können nur solche MLMs enthalten, deren benötigte Daten direkt beim Aufruf übergeben werden und die darüber hinaus keine zusätzlichen Daten lesen müssen. Für alle anderen MLMs besteht weiterhin das Problem der heterogenen Datenquell-Landschaft.

Auch bei Verwendung desselben Datenzugriffsverfahrens (z. B. SQL) können sich die Datenstrukturen in Institutionen

unterscheiden und erfordern eine Änderung der „Curly-Brace“-Ausdrücke in den MLMs. Institutionen mit unterschiedlichen Datenstrukturen, die MLMs untereinander austauschen wollen, müssten sich somit auf eine gemeinsame Sprache für die „Curly-Braces“ einigen. Die Frage des unterschiedlichen Zugriffs auf die Datenquellen wird somit in die Datenzugriffskomponenten verlagert und kann dort institutionsspezifisch implementiert werden.

Verfügbare Systeme

Der entwickelte Arden-Syntax-Standard als Grundlage für Ruleengine-Software sowie für praktische Anwendungen zur klinischen Entscheidungsunterstützung hat Eingang in eine Reihe von kommerziellen Produkten, wie z. B. von Agfa-Gevaert N.V., Allscripts, McKesson Corp., Siemens AG, unserer eigenen Firma Medexter Healthcare [8] und anderen gefunden. Neben diesen Closed-Source-Produkten, gibt es auch zwei dokumentierte Open-Source-Implementierungen eines Arden-Syntax-Compilers: Arden/J [9] und Arden2bytecode [10].

Beispiele des klinischen Einsatzes der Medexter Healthcare Arden-Syntax-Software

Der Arden-Syntax-Server von Medexter Healthcare

Medexter's Arden-Syntax-Server ist eine um SOA-Komponenten erweiterte Arden-Syntax-Ruleengine, die durch eine einfache Webservice-Schnittstelle in beliebige Systeme integriert werden kann. Auch hier sind Anpassungen an neue Systemumgebungen notwendig, die sich aber auf die Datenzugriffskomponenten beschränken. Der Server kann auch ohne Datenzugriffskomponenten betrieben werden, sofern die Daten den vorhandenen MLMs direkt während des Aufrufs übergeben werden. Solche MLMs können ohne technische Anpassungen (ggf. mit Ausnahme von numerischen Grenzwerten in klinischen Parametern und anderen medizinischen Kontexten) zwischen Institutionen ausgetauscht oder aber auch an einer zentralen Stelle hinterlegt und von mehreren Institutionen benutzt werden.

Eine strukturelle Darstellung des Zusammenspiels der Arden-Syntax-Softwarekomponenten von Medexter Healthcare mit unterschiedlichen Datenquellen ist in Abbildung 1 ersichtlich.

Hepaxpert – wissensbasierte Interpretation von Hepatitis-Serologie-Befunden

Die Hepaxpert-Wissensbasis [11], die aus einem Paket von mehreren MLMs besteht, ermöglicht eine präzise und ausführliche Interpretation von Hepatitis-Serologie-Testergebnissen. Dies erfolgt durch klinisch-relevante Interpretation der im Labor erhobenen Befunde. Die Wissensbasis von Hepaxpert enthält klinisch gut akzeptierte, in MLMs verpackte Interpretationsregeln, um alle möglichen – auch extrem seltene und komplexe – Kombinationen serologischer Testergebnisse zu interpretieren (siehe Abbildungen 2 und 3).

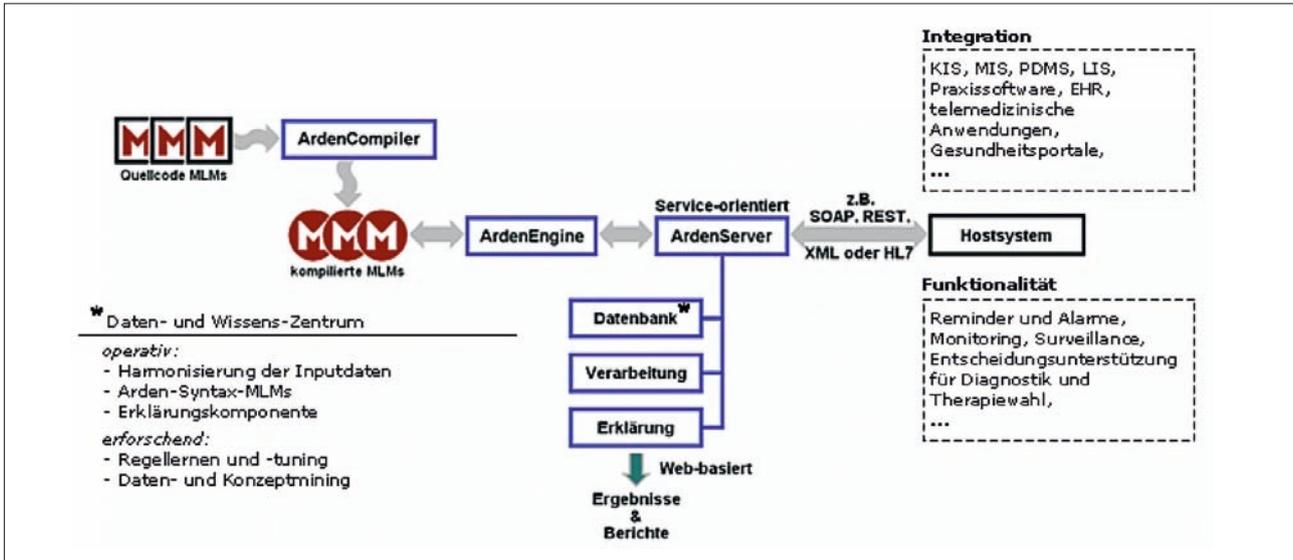


Abbildung 1: Arden-Syntax, Arden-Syntax-Server und medizinische Informationssysteme

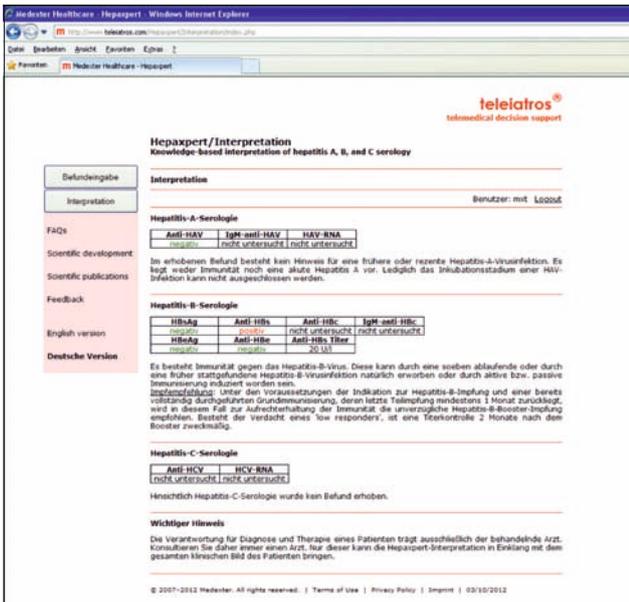


Abbildung 2: Automatisierte klinisch-orientierte Interpretation von Hepatitis-Serologie-Befunden durch Hepaxpert



Abbildung 3: Hepaxpert auf dem iPhone und iPad (beide in Englisch)

Moni-ICU – Monitoring und Reporting nosokomialer Infektionen auf Erwachsenen-Intensivstationen

Das System Moni-ICU [12, 13] dient der Früherkennung, dem kontinuierlichen Monitoring, der Surveillance und dem Reporting nosokomialer Infektionen. Die von der europäischen HELICS-Gruppe (Hospital in Europe Link for Infection Control through Surveillance) erarbeiteten Konsensuskriterien zur standardisierten Surveillance von nosokomialen Infektionen auf Intensivstationen, jetzt übernommen vom European Centre of Disease Prevention and Control (ECDC) [14], wurden unter Zuhilfenahme von wissenschaftlichen Methoden in einem schrittweisen Top-down-Verfahren in computerbasierte Regeln und Algorithmen überführt. Dabei wurde die den modellierten klinischen Begriffen innewohnende linguistische

Unschärfe mit Methoden der Fuzzy-Mengenlehre berücksichtigt [15].

Anschließend wurden die erstellten Algorithmen und Regeln als MLMs in der Arden-Syntax niedergeschrieben. Die dazugehörige Arden-Syntax-Ruleengine ist mit einer Datenbank gekoppelt, die die notwendigen Eingabedaten sammelt sowie Zwischen- und Endberechnungsergebnisse speichert. Beide – Datenbank und Ruleengine – residieren auf einem Server, der die entsprechenden Kopplungsmöglichkeiten zu den Datenquellen und Ausgaberroutinen bereitstellt. Dabei handelt es sich derzeit um intensivmedizinische Informationssysteme Philips CareVue (zukünftig ICIP) sowie das Laborinformationssystem der Mikrobiologie.

Moni-ICU wurde auf der oben beschriebenen Grundlage entwickelt, implementiert und ist am Klinischen Institut für Krankenhaushygiene am Allgemeinen Krankenhaus der Stadt Wien – Medizinischer Universitätscampus (AKH Wien) im Routineeinsatz. Täglich werden von 12 Intensivstationen mit 96 Betten alle für Moni-ICU relevanten Daten importiert, in der zur Arden-Syntax-Ruleengine zugehörigen Datenbank gespeichert und von den – derzeit 74 – MLMs schrittweise im Bottom-up-Verfahren verarbeitet. Dabei werden auch die in Moni-ICU enthaltenen computerbasierten Definitionen der Surveillance-Infektionen evaluiert. Die eingesetzte Fuzzy-Logik erlaubt nun die Bestimmung von voller, teilweiser oder nicht vorhandener Übereinstimmung der Patientendaten mit den entsprechenden Definitionen. Eine Darstellung des Moni-ICU-Surveillance-Schirmes ist in Abbildung 4 zu finden.

Integration von Arden-Syntax-Software in das Krankenhausinformationssystem i.s.h.med

Im Rahmen der Neuinstallation eines Krankenhausinformationssystems am AKH Wien wurden – sowohl für die klinische Routine als auch für Forschung und Lehre – Softwarekomponenten zur Entwicklung und dem Einsatz von Systemen zur klinischen Entscheidungsunterstützung integriert. Dem Informationssystem i.s.h.med der Siemens AG wurden im Entwicklungsbereich ein Arden-Syntax-Server sowie Entwicklungs- und Testsoftware (Arden-Syntax-IDE) beigelegt.

Ein weiterer Server dient der Qualitätssicherung im Produktionsbereich und ein dritter der eigentlichen Produktion. Eine entsprechende Strukturzeichnung ist in Abbildung 5 zu finden.

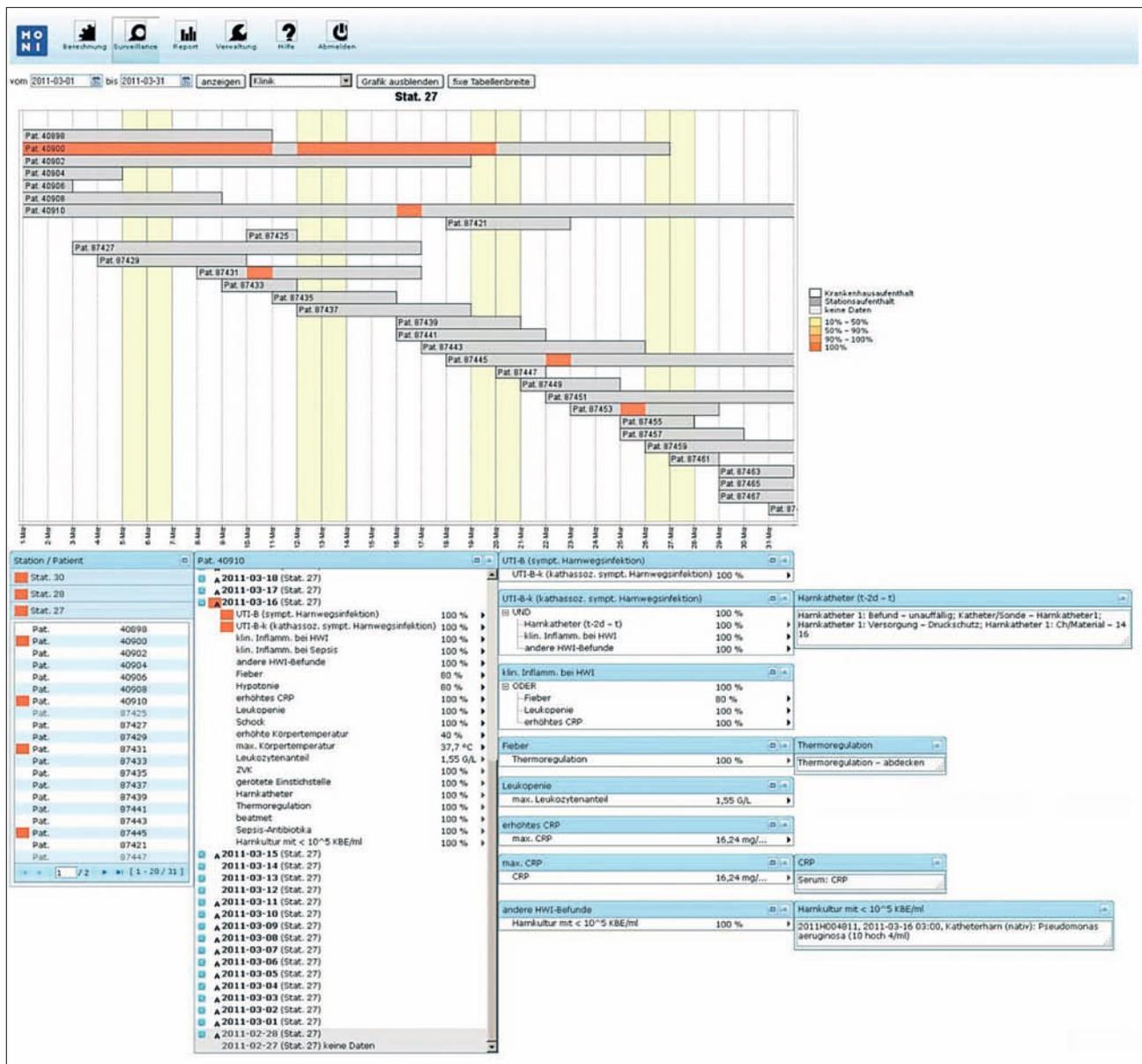


Abbildung 4: Wissensbasierte Früherkennung und automatisiertes Monitoring von nosokomialen Infektionen auf Intensivstationen mit Moni-ICU (Ergebnisse sind im unteren mittleren, Erklärungen im unteren rechten Teil zu finden; der obere Teil zeigt die Infektionsperioden im zeitlichen Verlauf)

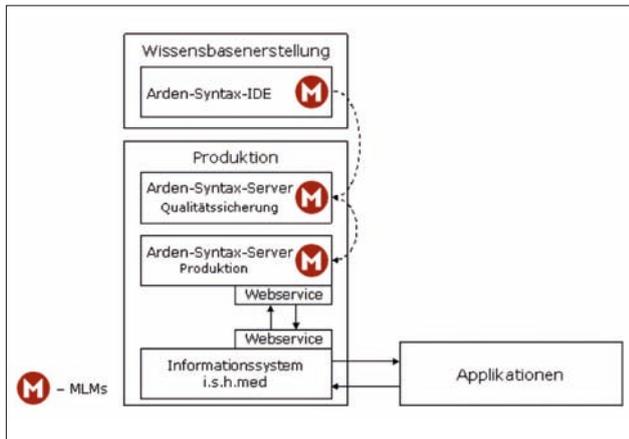


Abbildung 5: Integration der Arden-Syntax-Software in das Krankenhausinformationssystem i.s.h.med

Das Krankenhausinformationssystem i.s.h.med kommuniziert mit dem Arden-Syntax-Server über dessen Webservice-Schnittstelle, indem die zu analysierenden Daten an die entsprechenden MLMs, die auf dem Arden-Syntax-Server hinterlegt sind, gesendet werden. Die von den MLMs generierten Ergebnisse werden an das KIS zurückübertragen und dort in der Benutzeroberfläche des i.s.h.med angezeigt.

Die Universitätsklinik für Dermatologie des AKH Wien hat ein System zur Vorhersage von metastatischen Ereignissen bei Patienten mit Melanomen auf Basis der Arden-Syntax-Software im Testeinsatz [16]. Basierend auf dieser Plattform werden derzeit weitere Wissensbasen entwickelt und über den gleichen Mechanismus zugänglich gemacht. Für die Erweiterung, Erneuerung und Wartung der Wissensbasen wurde der in Abbildung 6 dargestellte Workflow etabliert.

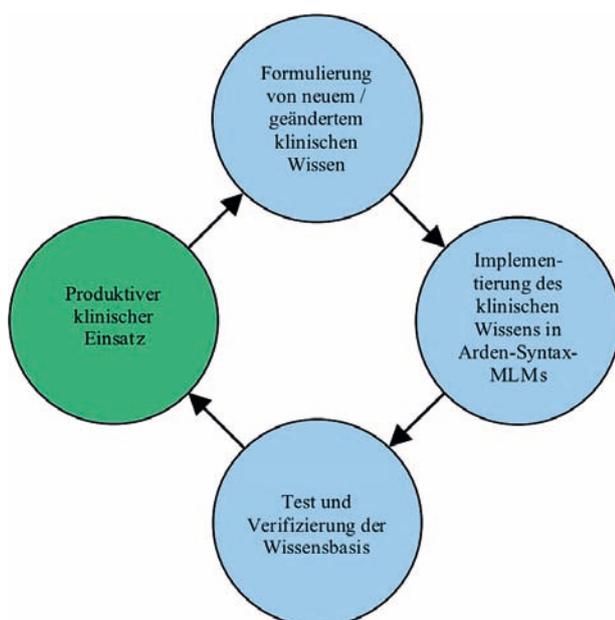


Abbildung 6: Workflow zur Erweiterung, Erneuerung und Wartung der Wissensbasen

Innovationsprojekt der Veterans Health Administration (VHA), USA

Ein kürzlich erfolgreich abgeschlossenes Projekt der Veterans Health Administration, USA, gemeinsam mit den Firmen Hewlett-Packard, USA, Medical Objects, Australien, und Medexter Healthcare [17] beruht auf einem ähnlichen Ansatz wie am AKH Wien, nur dass hier direkte Datenzugriffe über die Query and Expression Language GELLO [18], einem weiteren HL7-Standard, erfolgen sollten. Das Zusammenspiel zwischen Arden-Syntax und GELLO unter Einbeziehung eines Virtual Medical Records (vMR) [19] war erfolgreich und konnte in diesem Projekt zur Entwicklung klinischer Reminder und von Patientenanleitungen (patient report cards) genutzt werden.

Ausblick

Im Januar 2012 wurde die Version 2.8 der Arden-Syntax fertig gestellt und liegt zur Annahme der HL7-Administration vor. Im Jahre 2012 ist eine Fuzzifizierung der Arden-Syntax [20, 21] sowie eine Erweiterung der XML-Repräsentation geplant [22]. Diese Erweiterung der Syntax um Methoden der Fuzzy-Logik ist sinnvoll, da in täglichen klinischen Entscheidungen nicht immer mit scharfen Grenzwerten gearbeitet werden kann. Mittels Fuzzy-Mengen lassen sich z. B. erhobene klinische Parameter an den Bereichsgrenzen besser erfassen und bewerten.

In weiterer Folge soll, laut dem 3-Jahres-Plan der Arden-Syntax-Gruppe, das „Curly-Brace“-Problem mittels RIM (Reference Information Model)-basiertem Datenaustausch behoben werden.

Erste Ansätze zum Einsatz der Arden-Syntax bei telemedizinischen Anwendungen [23] sowie im Bereich der personalisierten Medizin [24] sind entwickelt worden.

Klaus-Peter Adlassnig (1, 2), Karsten Fehre (1)
 1 Medexter Healthcare GmbH
 Borschkegasse 7/5, A-1090 Wien, Österreich
 2 Institut für Medizinische Experten- und
 Wissensbasierte Systeme
 Zentrum für Medizinische Statistik, Informatik und
 Intelligente Systeme
 Medizinische Universität Wien
 Spitalgasse 23, A-1090 Wien, Österreich

Referenzen

- [1] Kawamoto, K., Houlihan, C.A., Balas, E.A., Lobach, D.F. (2005) Improving Clinical Practice Using Clinical Decision Support Systems: A Systematic Review of Trials to Identify Features Critical to Success. British Medical Journal 330 (7494): 765.
- [2] Health Level Seven International, Homepage: <http://www.hl7.org/>.
- [3] Arden Syntax Working Group, Homepage: <http://www.hl7.org/special/Committees/arden/index.cfm>.

- [4] Adlassnig, K.-P., Rappelsberger, A. (2008) Medical Knowledge Packages and Their Integration into Health-Care Information Systems and the World Wide Web. In Andersen, S.K., Klein, G.O., Schulz, S., Aarts, J., Mazzoleni, M.C. (Eds.) *eHealth Beyond the Horizon—Get IT There*. Proceedings of the 21st International Congress of the European Federation for Medical Informatics (MIE 2008), IOS Press, Amsterdam, 121–126.
- [5] Jenders, R.A., Corman, R., Dasgupta, B. (2003) Making the Standard More Standard: A Data and Query Model for Knowledge Representation in the Arden Syntax. In Musen, M. (ed.) *AMIA Annual Symposium Proceedings 2003*, American Medical Informatics Association, Washington, DC, 323–327.
- [6] Arden Syntax Work Group, Three-Year Plan, http://wiki.hl7.org/index.php?title=File:Arden_Syntax_WG_3y_Plan_2011-03-31.pdf.
- [7] Fehre, K., Adlassnig, K.-P. (2011) Service-Oriented Arden-Syntax-Based Clinical Decision Support. In Schreier, G., Hayn, D., Ammenwerth, E. (Eds.) *Tagungsband der eHealth2011 – Health Informatics meets eHealth – von der Wissenschaft zur Anwendung und zurück, Grenzen überwinden – Continuity of Care*, 26.–27. Mai 2011, Wien, Österreichische Computer Gesellschaft, Wien, 123–128.
- [8] Medexter Healthcare, Homepage: <http://www.medexter.com>.
- [9] Karadimas, H.C., Chailloleau, C., Hemery, F., Simonnet, J., Lepage, E. (2002) Arden/J: An Architecture for MLM Execution on the Java Platform. *Journal of the American Medical Informatics Association* 9, 359–368.
- [10] Gietzelt, M., Goltz, U., Grunwald, D., Lochau, M., Marscholke, M., Song, B., Wolf, K.-H. (2011) ARDEN2BYTECODE: A One-Pass Arden Syntax Compiler for Service-Oriented Decision Support Systems Based on the OSGi Platform. *Computer Methods and Programs in Biomedicine*, doi:10.1016/j.cmbp.2011.11.003.
- [11] Medexter Healthcare – Hepaxpert. Available: <http://www.medexter.com/hepax/hepax>.
- [12] Adlassnig, K.-P., Blacky, A., Koller, W. (2009) Artificial-Intelligence-Based Hospital-Acquired Infection Control. In Bushko, R.G. (Ed.) *Strategy for the Future of Health*, *Studies in Health Technology and Informatics* 149, IOS Press, Amsterdam, 103–110.
- [13] Blacky, A., Mandl, H., Adlassnig, K.-P., Koller, W. (2011) Fully Automated Surveillance of Healthcare-Associated Infections with MONI-ICU – A Breakthrough in Clinical Infection Surveillance. *Applied Clinical Informatics* 2 (3), 365–372.
- [14] European Centre for Disease Prevention and Control (ECDC) (2010) *European Surveillance of Healthcare-Associated Infections in Intensive Care Units, HAICU Protocol v1.01, Standard and Light*, 1–68, December 2010, Available: http://www.ecdc.europa.eu/en/aboutus/calls/Procurement%20Related%20Documents/5_ECDC_HAICU_protocol_v1_1.pdf.
- [15] Adlassnig, K.-P., Blacky, A., Koller, W. (2008) Fuzzy-Based Nosocomial Infection Control. In Nikraves, M., Kacprzyk, J., Zadeh, L.A. (Eds.) *Forging New Frontiers: Fuzzy Pioneers II – Studies in Fuzziness and Soft Computing* vol. 218, Springer, Berlin, 343–350.
- [16] Mehl, T., Binder, M., Scheibboeck, C., Holub, S., Adlassnig, K.-P. (2011) Integration of Clinical Decision Support into a Hospital Information System to Predict Metastatic Events in Patients with Melanoma. In Schreier, G., Hayn, D., Ammenwerth, E. (Eds.) *Tagungsband der eHealth2010 – Health Informatics meets eHealth – von der Wissenschaft zur Anwendung und zurück, Der Mensch im Fokus*, 6.–7. Mai 2010, Wien, Österreichische Computer Gesellschaft, Wien, 107–112.
- [17] VHA CHIO Innovation Project “Enhanced Clinical Decision Support for CPRS”, Homepage: <https://sites.google.com/site/enhancedcprscds>.
- [18] Sordo, M., Ogunyemi, O., Boxwala, A.A., Greenes, R.A. (2003) GELLO: An Object-Oriented Query and Expression Language for Clinical Decision Support. In Musen, M. (ed.) *AMIA Annual Symposium Proceedings 2003*, American Medical Informatics Association, Washington, DC, 1012.
- [19] Virtual Medical Record (vMR), [http://wiki.hl7.org/index.php?title=Virtual_Medical_Record_\(vMR\)](http://wiki.hl7.org/index.php?title=Virtual_Medical_Record_(vMR)).
- [20] Vetterlein, T., Mandl, H., Adlassnig, K.-P. (2010) Fuzzy Arden Syntax: A Fuzzy Programming Language for Medicine. *Artificial Intelligence in Medicine* 49(1), 1–10.
- [21] Vetterlein, T., Mandl, H., Adlassnig, K.-P. (2010) Processing Gradual Information with Fuzzy Arden Syntax. In Safran, C., Reti, S., Marin, H. (Eds.) *Proceedings of the 13th World Congress on Medical Informatics (MEDINFO 2010)*, *Studies in Health Technology and Informatics* 160, IOS Press, Amsterdam, 831–835.
- [22] Jung, C.Y., Sward, K.A., Haug, P.J. (2011) Executing Medical Logic Modules Expressed in ArdenML Using Drools. *Journal of the American Medical Informatics Association*, doi: 10.1136/amiajnl-2011.000512.
- [23] Fehre, K., Adlassnig, K.-P. (2010) Rule-Engine-Technologien als Innovationspotential für Telemedizin. In Böckmann, B., Semler, S.C., Dujat, C., Schug, S., Steyer, G. (Hrsg.) *Telemedizin – Erfolgsmodell für moderne Patientenversorgung: Tagungsband der TELEMED 2010*, 3.–5. November 2010, Berlin, TMF – Technologie- und Methodenplattform für die vernetzte medizinische Forschung e.V., Berlin, 64–69.
- [24] Samwald, M., Stenzhorn, H., Dumontier, M., Marshall, M.S., Luciano, J., Adlassnig, K.-P. (2011) Towards an Interoperable Information Infrastructure Providing Decision Support for Genomic Medicine. In Moen, A., Andersen, S.K., Aarts, J., Hurlen, P. (Eds.) *User Centred Networked Health Care – Proceedings of MIE 2011*, *Studies in Health Technology and Informatics* 169, IOS Press, Amsterdam, 165–169.