

Online-Datenbank für die Qualitätssicherung von Agrarprodukten am Beispiel der Kartoffel

SASCHA SLAVNIC, INSTITUT FÜR AGRARTECHNIK BORNIM E. V.
KLAUS GOTTSCHALK, INSTITUT FÜR AGRARTECHNIK BORNIM E. V.
ERHARD KONRAD, TU ZU BERLIN

Abstract

An advanced design of an online database of potatoes based on internet-technologie and the Client-Server model will be presented. The integration of different methods will be demonstrated.

1 Einführung

Im Rahmen eines mehrjährigen Projekts mit dem Ziel der Identifikation von qualitätsrelevanten Merkmalen der Kartoffel für die Vorhersage von Schwarzfleckigkeit, wurden und werden umfangreiche Datenbestände gesammelt. Damit die Daten effizient verwaltet werden, strebten wir den Einsatz einer Datenbank an, zumal mehrere Mitarbeiter die Daten parallel erfassen und auswerten sollten. Die unterschiedliche Hardware- und Softwareausstattung der einzelnen Mitarbeiter mußte ebenfalls berücksichtigt werden. Darüber hinaus wurde eine Online-Erfassung vor Ort mit Hilfe eines Kleincomputers mit Modem und eines Handys in Betracht gezogen.

2 Auswahl der Betriebsmittel

Der Einsatz einer zentralen Datenbank wurde aus mehreren Gründen angestrebt:

- die erfaßten Datensätze sind umfangreich; mehrere Mitarbeiter arbeiten gleichzeitig an der Erfassung
- das Projekt hat eine mehrjährige Dauer; die Datenbank muß ständig aktualisiert werden
- es sollen Teile der Datenbank in einem frühen Stadium ausgewertet werden, dabei soll immer der aktuellste Stand berücksichtigt werden
- die Datenerfassung und -verarbeitung außerhalb des Instituts für Agrartechnik Bornim e. V. sollte möglich sein

Im Institut für Agrartechnik existiert eine Vielzahl von unterschiedlichen Rechnerplattformen. Diesen verschiedenen Rechnerplattformen ist jedoch gemeinsam, daß sie alle über das TCP/IP-Protokoll kommunizieren können. Auf dem TCP/IP-Protokoll setzt das Hypertext Transport Protocol auf. Das HTTP erschien ein ideales Protokoll für unsere Anwendung zu sein, denn die Hypertext-Markup-Language, die von jedem Web-Browser zur Darstellung von Internet-Seiten benutzt wird, setzt auf dem HTTP auf. HTML ermöglicht über den FORM-Mechanismus Interaktion mit dem Web-Server. Das heißt, daß Formulare im Web-Browser ausgefüllt werden und an den Web-Server abgesandt werden. Wir entschlossen uns daher, den Zugriff auf die Datenbank über einen Web-Server zu realisieren. Als Hardware wurde ein normaler PC verwendet. Linux wurde als

Betriebssystem, Adabas D als Datenbankverwaltungssystem und Apache mit SSL-Erweiterung als Web-Server verwendet.

3 System-Layout

Das System gliedert sich in drei Programmteile:

- Tabellenkalkulations-Import-Modul
- Datenbankerfassungs- und -bearbeitungs-Modul
- Tabellenkalkulations-Export-Modul

Systemarchitektur

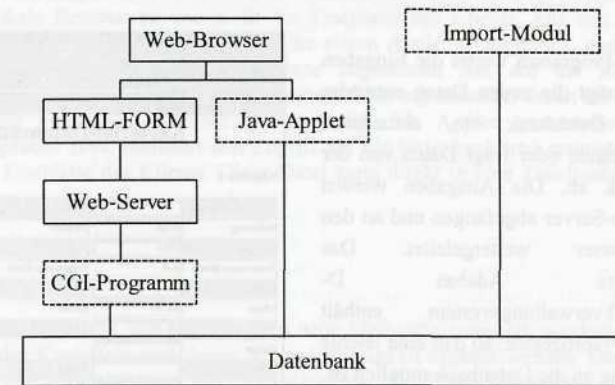


Bild 1 - Darstellung des Import-, Export-, und Bearbeitungs-Moduls anhand der Systemarchitektur

Das Import-Modul ist ein eigenständiges Programm und greift über die Adabas D - ODBC-Schnittstelle auf die Datenbank zu. Dieser Import-Vorgang ist nur einmal zum Übertragen der Daten in die Datenbank notwendig. Danach wird das Programm nicht mehr benötigt.

Das Datenbearbeitungsmodul braucht genauso wenig auf den Benutzer-Rechner (Client) installiert zu werden, wie das Export-Modul, da beide Programme über den Web-Browser geladen, bzw. dargestellt werden.

3.1 Tabellenkalkulations-Import-Modul

Zu Beginn des Projekts, als der Umfang sich noch nicht im vollen Maß abschätzen ließ, wurden die Meßdaten in einer Tabellenkalkulation erfaßt. Als die Datenmenge zu umfangreich wurde, und der Einsatz einer Datenbank zweckmäßig erschien, standen wir vor dem Problem der Datenkonvertierung. Eine der Alternativen wäre die manuelle Neuerfassung gewesen, die jedoch aus Zeit- und Kostengründen ausschied. Wir entwickelten ein Programm, mit Hilfe der Programmierungsumgebungen Delphi und Amzi

Prolog, das per DDE¹ die Daten direkt aus der Tabellenkalkulation liest. Eine Inferenzmaschine analysiert die Struktur der Daten. Die Strukturanalyse ist weitgehend automatisiert, es ist nur an wenigen Stellen eine Korrektur des Benutzers erforderlich. Die erzeugte Datenbankbeschreibungdatei muß anschließend normalisiert werden, jedoch wird der Vorgang vom System unterstützt. Nachdem die Datenbank aufgesetzt wurde, können die Daten aus der Tabellenkalkulation in die Datenbank per DDE übertragen werden.

3.2 Datenbankerfassungs- und -bearbeitungs-Modul

Diese Komponenten wurden als sogenannte CGI²-Programme realisiert. CGI-Programme werden vom Web-Server gestartet und erhalten die Eingaben des Web-Browsers.

Das CGI-Programm wertet die Eingaben aus und trägt die neuen Daten entweder in die Datenbank ein, aktualisiert Datenbestände oder fragt Daten von der Datenbank ab. Die Ausgaben werden vom Web-Server abgefangen und an den Web-Browser weitergeleitet. Das verwendete Adabas D-Datenbankverwaltungssystem enthält einen C-Präprozessor, so daß eine leichte Anbindung an die Datenbank möglich ist.

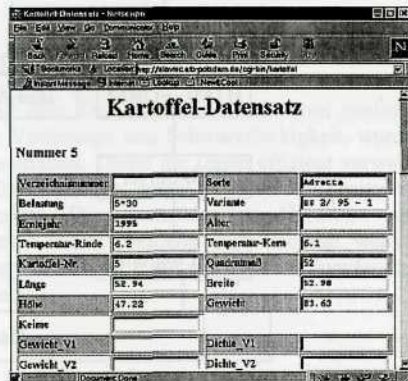


Bild 2 Web-Browser auf einem PC

Die Daten werden im Web-Browser in Formularen visualisiert und bearbeitet. Die HTML-Sprache verfügt über Komponenten wie Schalter, Textfelder, Radio-Schalter und Eingabefelder.

Selbst auf einem Kleincomputer wie dem 3COM Palm Pilot läuft ein Web-Browser, der für die Datenbankerfassung und -bearbeitung geeignet ist (Bild 3).



Bild 3 Datenverarbeitung mit einem Kleincomputer

3.3 Tabellenkalkulations-Export-Modul

Zur Weiterverarbeitung der erfaßten Datenbestände müssen die Daten aus der Datenbank in ein für Tabellenkalkulationen verständliches Format überführt werden. Als geeignetes Format kann das Data Interchange Format (kurz DIF) angesehen werden, daß von gängigen Tabellenkalkulationen wie Excel und Lotus unterstützt wird.

Ein größeres Problem warf der Transfer der DIF-Datei vom Server zum Client, also zum jeweiligen Benutzer-Rechner auf. Die DIF-Datei auf dem Server zu erstellen und eine HTML-Seite zum Herunterladen der Datei zu erzeugen, erschien als nicht zweckmäßig, da dem Server unnötige Arbeit aufgebürdet würde und ein nicht hundertprozentig zuverlässiger Caching-Mechanismus hätte verwendet werden müssen. Die Abfrage der Datenbank und die Speicherung der DIF-Datei lokal auf dem Client durch den Benutzer-Rechner ist die optimale Lösung. Die Restriktionen des Web-Browsers, die keine Aktionen außerhalb der sogenannten **Sandbox** zulassen, können durch den Einsatz eines gesicherten SSL-Web-Servers aufgehoben werden. Das Sandbox-Sicherheitskonzept verhindert den Zugriff auf lokale Ressourcen wie z. B. die Festplatte des Clients. Für den Netscape Communicator gibt es jedoch Java-Klassen, die einem Applet ermöglichen, nachdem der Benutzer der Erhöhung der Sicherheitsklasse zugestimmt hat, auf die Ressourcen zuzugreifen. Voraussetzung ist jedoch eine Authentifizierung entweder durch ein signiertes Jar-Archiv oder einen SSL-Web-Server notwendig. Ein Applet, geschrieben in der Programmiersprache Java, realisiert den Zugriff auf die Datenbank und erzeugt die DIF-Datei auf der Festplatte des Clients. Diese Datei kann direkt in eine Tabellenkalkulation geladen werden.

4 Ausblick

Die bestehende Datenbank soll nicht nur um neue Meßwerte erweitert, sondern auch um Abbildungen der Kartoffeln und deren Schwarzfleckigkeit ergänzt werden. Das Export-Modul wurde nur entwickelt, um eine Visualisierung und Weiterbearbeitung durch eine Tabellenkalkulation zu ermöglichen. Ein Teil der Visualisierungsmöglichkeiten sollen in das bestehende System eingebettet werden, damit die Arbeitsschritte „DIF-Datei exportieren“, „Tabellenkalkulation starten“ und „Daten in der Tabellenkalkulation weiterverarbeiten“ entfallen können und sich die Handhabung des Systems optimaler gestaltet. Das System dient als Basis für ein Wissensbasiertes System zur Vorhersage von Schwarzfleckigkeit bei Kartoffeln.

Literatur

- ULLMAN, J. (1998): Database And Knowledge-Base Systems, Vol. 1 + 2. Computer Science Press
- NETSCAPE (1998): <http://developer.netscape.com/security>

¹ Direct Data Exchange

² Common Gateway Interface