Zeitschrift: Bulletin Electrosuisse
Herausgeber: Electrosuisse, Verband für Elektro-, Energie- und Informationstechnik
Band: 94 (2003)
Heft: 1

Artikel: Services web pour systèmes embarqués
Autor: Gabioud, Dominique
DOI: https://doi.org/10.5169/seals-857506

Nutzungsbedingungen
Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation
L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use
The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 05.07.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

Services web pour systèmes embarqués

L'émergence de la technologie des services web (web services) est en train de changer en profondeur l'architecture des systèmes de communication. Des environnements comme Microsoft.NET ou SunONE sont largement basés sur les services web, par ailleurs en cours de standardisation par le World Wide Web Consortium[1] (W3C). Aujourd'hui des systèmes embarqués avec des coûts de production peu élevés sont suffisamment puissants pour permettre l'exécution de services web. Le projet web-enabled-devices financé par le programme de la Confédération Soft[net][2] dans le cadre du réseau national EdiSoN[3] a pour but d'obtenir l'expertise nécessaire au développement de services web dans un environnement à ressources limitées – c'est-à-dire: faible coût, basse consommation, petite taille.

De sa naissance à aujourd'hui, l'internet a raccordé des équipements de plus en plus petits et de plus en plus nombreux: des mainframes d'abord, des PC ensuite et, plus récemment, des équipements mobiles (téléphones mobiles, PDA[4]). La dernière étape de cette évolution consiste à connecter les dispositifs (*devices*) les plus divers. Par «dispositif», on entend tout équipement qui mesure ou modifie

Dominique Gabioud

des grandeurs physiques dans son environnement: une machine à café, un capteur de température, une machine-outil, une station météo, etc.

Aujourd'hui, ces dispositifs fonctionnent de manière autonome ou sont reliés à un ordinateur par une connexion directe. Puisqu'ils sont déjà équipés de micro-contrôleurs, une connexion TCP/IP[5] n'engendre qu'un coût marginal faible, acceptable pour beaucoup d'applications.

L'internet est principalement utilisé pour des communications de type homme–machine (telnet, FTP[6], web) ou homme–homme (courrier électronique). La communication machine–machine exige un formalisme beaucoup plus strict. Des initiatives telles que le *semantic web*[7] [1] ou les *web services*[8] [2, 3] ont précisément comme objectif de définir un tel formalisme. Ce type de communication revêt une importance particulière pour la connexion de dispositifs embarqués: du fait de leur grand nombre, ces dispositifs communiqueront certainement avec d'autres dispositifs et non directement avec une personne.

La technologie Ethernet TCP/IP a démontré sa capacité à servir de support universel pour tous les services de télécommunication (la téléphonie, l'interconnexion sécurisées de réseaux locaux LAN, la communication industrielle pour le contrôle de processus, etc.). Les réseaux issus de cette technologie supplantent de plus en plus des réseaux dédiés. Dans ce contexte, les dispositifs doivent avoir directement une connectivité compatible avec les standards web et TCP/IP pour devenir des éléments à part entière d'un système d'information. Toutefois, implémenter un serveur HTTP[9] délivrant des documents HTML[10] n'est pas suffisant car l'information offerte n'est pas structurée.

La connectivité des systèmes embarqués aujourd'hui

L'interface de communication des systèmes embarqués est souvent réalisée par l'intermédiaire d'une liaison série ou d'une connexion à un réseau de terrain.

Beaucoup de systèmes ont une interface série (de type RS 232 ou RS 485) sur laquelle un protocole propriétaire a été défini. La liaison est de type point à point et le protocole propriétaire doit être implémenté dans l'ordinateur directement raccordé au dispositif embarqué.

Les réseaux de terrain offrent certes des solutions de connexion mais celles-ci sont limitées au contexte particulier de l'automatisation de processus et ne s'intègrent pas facilement dans un système d'information global basé sur le web. Il est significatif de constater que les standards de réseaux de terrain, traditionnellement très hétérogènes et incompatibles entre eux, adoptent de plus en plus les standards universels que sont devenus Ethernet et TCP/IP.

Qu'est-ce qu'un service web?

L'architecture des services web a fait l'objet de nombreuses publications [2, 3]. Seuls les éléments de base sont brièvement récapitulés.

Le web a bâti son succès à partir de standards très simples mais permettant de construire des applications sophistiquées pour la communication homme–machine. Les services web appliquent la même philosophie pour essayer d'obtenir le même succès dans le domaine de la communication machine–machine.

Le *W3C Web Services Architecture group*[7] a donné la définition suivante d'un service web (les principaux éléments de cette définition sont décrits par la suite):

Un service web est une application logicielle identifiée par une URI (Uniform Resource Identifier) dont les interfaces et les liaisons peuvent être définies, décrites et découvertes par des données XML (Extensible Markup Language). Un service web supporte des interactions directes avec d'autres applications logicielles par l'utilisation de messages basés sur XML transportés par des protocoles de l'internet[11].

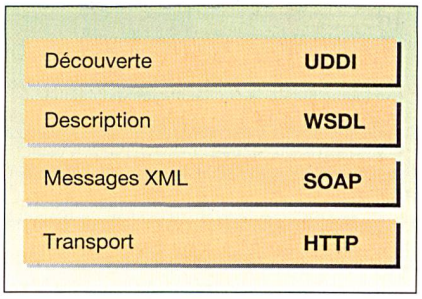

Figure 1 L'ensemble des mécanismes intervenant dans les services web

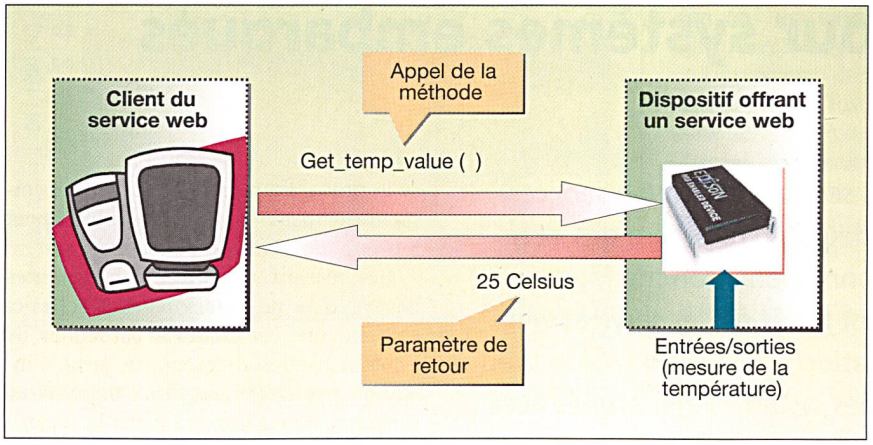

Figure 2 Le dispositif embarqué offre un service web
Un tel dispositif peut également utiliser des services web comme client

XML est à la base des services web

XML[12] [4] est un langage textuel créé pour décrire des données structurées. La structure d'un document XML peut être définie librement (par exemple dans un document XML appelé «XML schema»). XML partage avec HTML l'utilisation de balise pour marquer la structure du document, mais, contrairement à HTML, XML ne s'occupe pas de présentation des données.

Un service web est identifié par un URI

Des URI sont utilisés pour identifier toutes les ressources disponibles sur le web. Les services web ne font pas exception à cette règle. Généralement, les URI (par exemple http://ws.hevs.ch/web_service_name) sont composés de trois parties:

- le mécanisme d'accès à la ressource (http);
- le nom de la machine offrant la ressource (ws.hevs.ch);
- le chemin d'accès local à la ressource (/web_service_name).

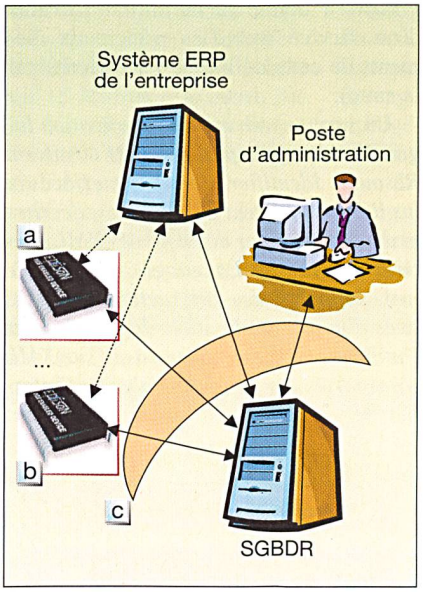

Figure 3 Gestion des utilisateurs par un poste administrateur
a: contrôle d'accès (entrée dans une salle, accès à une armoire de vestiaire, etc.); b: consommation de ressources (photocopieuse, automate à boisson, etc.); c: interface homogène basée sur les services web; ERP: Enterprise Resource Planning; SGBDR: Système de Gestion de Base de Données Relationnelles

Un service web peut être défini, décrit et découvert

Le langage WSDL[13] est une grammaire XML utilisée pour spécifier les propriétés d'un service web telles que les méthodes qu'il implémente et la manière de les appeler ou l'URI du service. La description WSDL d'un service peut être considérée comme une forme de contrat entre un service web et ses clients.

Des mécanismes[14] permettant de définir et de découvrir des services web existent mais sont encore peu répandus. Ils permettent la mise en place d'annuaires de services web avec des fonctions de moteur de recherche pour trouver la description d'un service à partir de certains éléments.

Les protocoles de l'internet sont utilisés pour le transport des messages

Selon la définition du W3C, des protocoles tels que FTP ou SMTP[15] pourraient être utilisés pour le transport de messages. En pratique, les messages sont transportés par HTTP et SOAP[16]. Fig. 1 résume l'ensemble des mécanismes intervenant dans les services web.

Un service web se présente sous la forme d'un ensemble de méthodes (au sens de la programmation) qui peuvent être appelées à distance[17] par un client. La fig. 2 présente l'exemple d'un dispositif embarqué qui mesure une température et la rend disponible à travers la méthode *Get_ temp_value()* du service web.

Services web pour systèmes embarqués – un exemple

L'exemple suivant permet de comprendre dans quel contexte l'utilisation de services web sur des systèmes embarqués peut être intéressante.

Une entreprise développe et vend des systèmes de monétique et de contrôle d'accès pour des environnements fermés (un hôtel, un hôpital, un stade, etc.). L'architecture du système est présentée dans la fig. 3.

Autour d'un *Système de Gestion de Base de Données Relationnelles* (SGBDR), différents périphériques permettent d'accéder à un local, de retirer une marchandise à un automate, de payer une consommation à la cafétéria, etc. Un poste administrateur permet de faire la gestion des utilisateurs. Le SGBDR est également en relation avec le système de gestion global Enterprise Resource Planning (ERP) de l'entreprise.

Le SGBDR communique avec tous les éléments (embarqués ou non) de manière uniforme, sur la base de services web. Chaque élément offre un ou plusieurs services web qui sont directement disponibles dans le SGBDR. Un élément du système peut également fonctionner comme client et recourir à des services web implémentés dans le SGBDR.

Dans le contexte des services web, les développeurs peuvent se concentrer sur les aspects fonctionnels (logique métier), le formatage et l'analyse des messages étant gérés automatiquement.

Intégrer des services web basés sur des standards stables et implémentés par tous les acteurs du monde informatique dans les éléments périphériques présente deux avantages annexes non négligeables:

- les éléments périphériques peuvent s'intégrer directement dans l'environnement informatique de l'entreprise (sans le SGBDR dédié) car les applicatifs en usage peuvent communiquer directement avec ces équipements;
- l'entreprise qui développe les systèmes de monétique et de contrôle d'accès n'est plus dépendante du type ou de la version du SGBDR. Il n'est ainsi plus nécessaire d'adapter ou de réécrire des *drivers* pour accéder aux périphériques.

Le système embarqué est vu comme un élément capable d'exécuter des méthodes qui peuvent être appelées à travers un réseau et/ou capable d'appeler des méthodes distantes. Ces capacités sont suffisamment génériques pour répondre à la

```
<?xml version="1.0" encoding="utf-8"?>
<definitions  name="AddService"
              xmlns="http://schemas.xmlsoap.org/wsdl/"
              xmlns:wsdlns="http://AddService.org/wsdl/"
              xmlns:xsd="http://www.w3.org/2001/XMLSchema"
              xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
  <service name="Add">
     <documentation>Add two numbers</documentation>
     <port name="AddPort" binding="wsdlns:AddBinding">
        <soap:address location="http://ws.hevs.ch:8080/add_s"/>
        <input name="addRequest">
           <message name="AddRequest">
              <part name="a" type="xs:int"/>
              <part name="b" type="xs:int"/>
           </message>
        </input>
        <output name="addResponse">
           <message name="AddResponse">
              <part name="sum" type="xs:int"/>
           </message>
        </output>
     </port>
  </service>
</definitions>
```

Figure 4 Description WSDL du service AddService

plupart des besoins de communication de systèmes embarqués. L'utilisation de services web est particulièrement intéressante lorsque les conditions suivantes sont remplies:

- le dispositif est connecté de manière permanente au web;
- plusieurs dispositifs doivent être reliés à un serveur qui n'est pas situé dans leur proximité immédiate;
- les dispositifs sont susceptibles d'être raccordés à des systèmes informatiques de type différent.

Développement de services web sur des systèmes embarqués

Les services web sont généralement présentés dans le cadre d'architectures telles que Microsoft.NET ou SunONE. Celles-ci requièrent des ressources matérielles de plusieurs ordres de grandeur supérieures à ce qui peut être disponible dans un système embarqué.

Les systèmes embarqués doivent répondre à des exigences sévères: faible prix de revient, faible consommation d'énergie, petite taille. Des services web peuvent néanmoins être implémentés dans de tels environnements pour autant qu'il soit fait un usage optimal des ressources disponibles.

Les méthodes d'un service et les paramètres qui y sont associés sont d'abord décrits selon le formalisme WSDL.

A titre d'exemple, la description WSDL du service *AddService* est présentée dans la fig. 4.

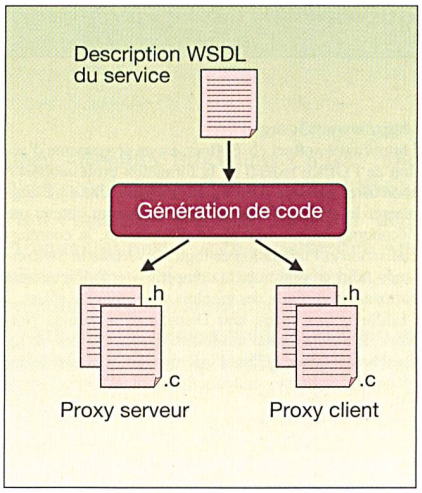

Figure 5 Génération automatique de proxies client et serveur

Dans cette description, on reconnaît l'URI du service (http://ws.hevs.ch:8080add_s), le message de requête (*AddRequest*) et les paramètres qui y sont associés (les entiers *a* et *b*) et le message de réponse (*AddResponse*) contenant l'entier *sum*.

Cette description est utilisée par un outil pour générer soit le squelette du code des méthodes du service web – *service proxy* – ou la méthode locale au client qui appelle le service web – *client proxy* – (fig. 5).

Pour développer un service web, il suffit d'écrire les méthodes du service et de les intégrer au *proxy* serveur et à la bibliothèque SOAP (fig. 6).

Du côté client, la description WSDL permet d'accéder aux méthodes du service comme si elles étaient disponibles localement (fig. 7).

En résumé, la problématique de la distribution est gérée de manière transparente pour les développeurs: les méthodes des services web sont développées comme des méthodes appelées localement et les clients appellent ces méthodes comme ils appelleraient des méthodes locales. La simplicité et l'efficience de cette méthode de développement n'est pas contradictoire avec une utilisation optimale des ressources matérielles disponibles.

La sécurité

La sécurité est une préoccupation de premier plan lorsque des dispositifs peuvent être contrôlés à travers un réseau, à fortiori lorsque ce réseau est l'internet lui-même.

L'utilisation d'une connexion HTTPS[18] apporte les éléments de sécurité suivant:

- authentification mutuelle du service web et de son client;
- confidentialité et intégrité des données transportées entre le service et le client.

Dans le cadre du projet *Secure Communication in Distributed Embedded Systems* – un autre projet EdiSoN[2] soutenu par Soft[net] – un coprocesseur de sécurité hardware est développé et intégré à une bibliothèque SSL/TLS[19]. Cette librairie et son coprocesseur de sécurité seront utilisés dans le cadre du projet afin de minimiser les ressources supplémentaires nécessaires à l'exécution de SSL/TLS.

L'utilisation d'une bibliothèque SSL/TLS purement logicielle est également

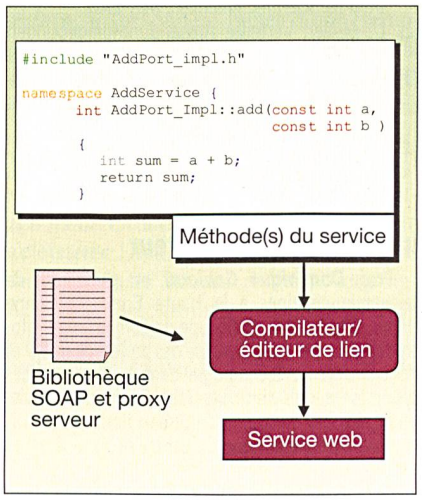

Figure 6 Méthode de développement pour un service web

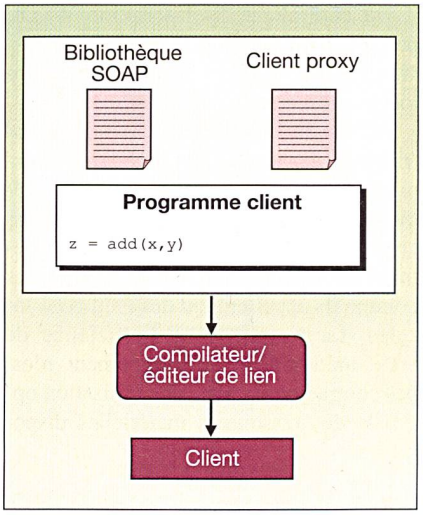

Figure 7 Méthode de développement pour un client

possible pour autant que l'on accepte les restrictions suivantes:

- un long temps d'établissement de la première connexion: SSL/TLS a un mécanisme de reprise qui raccourcit notablement le temps d'établissement des connexions suivantes;
- l'utilisation d'un ensemble réduit d'algorithmes de chiffrage: SSL/TLS permet aux deux partenaires de la communication de négocier l'utilisation d'algorithmes de chiffrage. Seuls des algorithmes les moins gourmands en ressources peuvent être implémentés sur le dispositif embarqué.

Enfin, XML comprend également ses propres mécanismes de sécurité pour le chiffrage, la gestion de clés et la signature numérique [2].

Referenzen

[1] *T. Berners-Lee, J. Hendler, O. Lassila:* The Semantic Web. Scientific American, May 2001
[2] *U. Müller:* Verteilte Anwendungsumgebungen mit XML-Web-Services. Bulletin SEV/ASE 19/02
[3] Unraveling the Web Services Web – An introduction to SOAP, WSDL and UDDI. IEEE Internet Computing March–April 2002
[4] *F. Jung:* XML – Grundstein zum grenzenlosen Datenaustausch. Bulletin SEV/ASE 09/00

Informations sur l'auteur

Prof. ***Dominique Gabioud***, est professeur de télécommunications à la Haute Ecole Valaisanne (HEVs). Il fait partie du Groupe de Compétences Infotronique, une unité qui réalise des projets de Ra&D et de transfert de technologie dans le domaine des systèmes intégrés distribués. Prof. Gabioud est également membre du comité du réseau National EDiSoN et du comité scientifique du Centre de Compétences en Technologies de l'Information de la Haute Ecole Spécialisée de Suisse Occidentale (HES-SO). *Coordonnées: Haute Ecole Valaisanne, CH-1950 Sion, dominique.gabioud@hevs.ch*

Web-enabled-devices

Un projet du réseau national EdiSoN

Le projet web-anbled-devices poursuit trois objectifs:

- acquérir un savoir-faire sur le développement de services web pour systèmes embarqués (web-enabled-devices, modules web)
- acquérir un savoir-faire dans l'intégration de modules web dans des environnements standard de systèmes d'information d'entreprise tels que Microsoft.NET ou SunONE
- diffuser le know-how acquis vers les étudiants des écoles d'ingénieurs et vers l'industrie.

Trois institutions collaborent à ce projet:

- La Haute Ecole Spécialisée de Suisse Occidentale – Haute Ecole Valaisanne (HEVs), groupe de compétences infotronique – assume le responsabilité générale du projet. Elle développe des systèmes embarqués supportant des services web sur la base de processeurs 32 bits (ARM 7 TDMI) et 8 bits dans un environnement natif (C, C++). Les systèmes d'exploitation utilisés sont ecos[20] et µClinux[21]. La méthode de développement pour services web et client définie ci-dessus a été validée sur cette plateforme avec les deux systèmes d'exploitation.
- La Fachhochschule Zentralschweiz – Hochschule für Technik und Architektur Luzern (HTA-L) – développe des services web sur la base de processeurs 8 bits (NEC 78K0, Motorola 68HC08) avec une machine virtuelle Java.
- La Fachhochschule Nordwestschweiz (Fachhochschule Solothurn, FHSO) traite les aspects d'intégration des modules web dans les environnements IT standards.

[1] http://www.w3c.org
[2] http://www.softnet.ch Softnet est un programme d'action de l'Office fédéral de la formation professionnelle et de la technologie (OFFT). Le but recherché est d'augmenter le chiffre d'affaires et le rendement obtenu par l'économie suisse avec le développement, la commercialisation et l'utilisation de logiciels «Made in Switzerland», afin de renforcer la compétitivité de l'économie nationale et de créer des emplois sur le marché suisse.
[3] EdiSoN: Embedded and Distributed Solutions Network. Réseau national qui fédère des institutions de recherche appliquée et développement actives dans le domaine des systèmes embarqués distribués. http://www.edison.ch
[4] PDA: Personal Digital Assistant
[5] TCP/IP: Transmission Control Protocol/Internet Protocol
[6] FTP: File Transfer Protocol
[7] http://www.w3c.org/2001/sw/
[8] http://www.w3c.org/2002/ws/
[9] HTTP: HyperText Transmission Protocol
[10] HTML: Hypertext Markup Language
[11] http://www.w3.org/TR/2002/WD-wsa-reqs-20021011
[12] http://www.w3.org/XML
[13] WSDL: Web Services Description Language
[14] UDDI: Universal Description, Discovery and Integration of Web Services. http://www.uddi.org
[15] SMTP: Simple Mail Transfer Protocol
[16] SOAP: Simple Object Access Protocol
[17] RPC: Remote Procedure Call
[18] HTTPS: HyperText Transport Protocol Secure. HTTP sur SSL/TLS. The TLS Protocol Version 1.0 (RFC 2246); HTTP Over TLS (RFC2818)
[19] SSL/TLS: Secure Sockets Layer/Transport Level Security.
[20] http://sources.redhat.com/ecos
[21] http://www.uclinux.org

Web-Services für Bordsysteme

Seit dem Aufkommen der «Web Services»-Technologie ist die Architektur von Kommunikationssystemen einem grundlegenden Wandel unterworfen. Umgebungen, wie z.B. Microsoft.NET oder SunONE, sind weit gehend auf Web-Services basiert, und die Normung durch das World Wide Web Consortium (W3C) ist gegenwärtig in Vorbereitung. Heute sind kostengünstige Bordsysteme verfügbar, deren Leistungsfähigkeit den Anforderungen solcher Webdienste durchaus gewachsen ist. Das vom Bundesprogramm Soft[net] im Rahmen des nationalen Netzes EdiSoN finanzierte Projekt «Web-Enabled-Devices» hat zum Ziel, die zur Entwicklung von Webdiensten in einer bescheidenen Umgebung – d.h. niedrige Kosten, geringer Verbrauch, kleine Abmessungen – notwendigen Kenntnisse zu erlangen.