

Versionshinweise zu Adobe® AIR™ 1.1

6/16/2008

Inhalt

Systemanforderungen für Adobe AIR 1.1	2
Funktionen von AIR 1.1	3
Änderungen am Application Descriptor	3
API-Ergänzungen in AIR 1.1	4
Zertifikatmigration	6
Installation von Adobe AIR	6
Installation der AIR 1.1 Runtime und der AIR-Anwendung	6
Nahtlose Installation.....	6
Ablaufdatum für die AIR Beta Runtime.....	6
Tipps und Tricks	6
Öffnen einer AIR-Anwendung mit Safari auf Mac OS	6
Einstellen des MIME-Typs auf Ihrem Webserver für AIR-Anwendungen.....	6
adl - Debugging-Support.....	6
Fehlerbehebung (für Endanwender)	7
PDF-Inhalte werden nicht angezeigt.....	7
SQL-Datenbank und Mac OS 10.5 Time Machine-Funktion	7
AIR-Anwendungen und MobyDock.....	7
Firefox wird nicht als Standardbrowser verwendet.....	7
Fehlerbehebung (für Entwickler)	7
HTTP-Beschränkungen bei Windows.....	7
Links zu Bildern werden nicht richtig geladen	7
Verwendung der Dateneigenschaft von NativeMenuItem	7
Nicht unterstützte URL-Schemata	7
Eigenschaft ContextMenu.visible.....	8
Unterstützung für wenig verbreitete Schriftzeichen für den China National Standard GB 18030-2000	8
Virtuelle Tastatur in SWF-Inhalt von HTML auf Tablet PC geladen	8
Windows-64-Bit-Betriebssystem: Installation von AIR-Anwendungen im 64-Bit-Programmordner	8
app: URL-Schema mit offenen PDF-Parametern auf dem Mac	8
htmlText-Links auf Vista.....	8
JavaScript-Datumsobjekte, die eine Sandbox Bridge passieren, verlieren ihren ursprünglichen Typ	8
Die Einstellung der Breite oder Höhe eines iframe auf 100 % kann unerwartete Ergebnisse verursachen	8
Die Spaltenaffinitäten DATE und DATETIME sind mit AIR Beta-Versionen nicht kompatibel	8

Das Verhalten der Spaltenaffinität ist mit AIR Beta-Versionen nicht kompatibel	9
Dock- und Menüleiste werden im Vollbildmodus bei der Überlagerung von Fenstern in Mac OS nicht ausgeblendet.....	9
NativeWindow.height wird im Resize-Ereignis falsch gemeldet.....	9
Festlegen von idleThreshold.....	9
Tracing zur Konsole	10
Weitergabe des Headers "Cookie" an URLRequest.requestHeaders.....	10
Vollständiger Zertifikatpfad für die Code-Signierung erforderlich.....	10
Von der Methode NativeWindow.globalToScreen() zurückgegebenes Punktobjekt kann auf Sekundärbildschirmen einen falschen Y-Wert haben.....	10
setAsDefaultApplication() und Dateierweiterungsverknüpfung unter Windows	10
removeAsDefaultApplication() und Dateierweiterungsverknüpfung unter Windows Vista und Mac	10
ADT-Verwendung des Proxys mit Authentifizierung.....	10
Registrierung des Dock-Symbolereignisses	10
Breite und Höhe für NativeWindow.....	10
Aktualisieren von Namen/Werte-Paaren im verschlüsselten lokalen Speicher.....	10

Systemanforderungen für Adobe AIR 1.1

Adobe hat Adobe® AIR™ 1.1 auf den folgenden Mindesthardwarekonfigurationen getestet:

Windows

- Intel® Pentium® III 1 GHz oder schnellerer Prozessor
- Microsoft® Windows® XP® mit Service Pack 2, Windows XP Tablet PC Edition, Windows Vista® Home Premium, Business, Ultimate oder Enterprise (einschließlich 64-Bit-Versionen), Windows 2000 mit Service Pack 4, Windows 2003 Server
- 512 MB RAM

Mac OS X

- Intel Core™ Duo 1,83 GHz oder schnellerer Prozessor, PowerPC® G4 1 GHz oder schnellerer Prozessor
- Mac® OS X® 10.4.11 oder Mac OS X 10.5.2
- 512 MB RAM

Sprachversionen

Die Installation von Adobe AIR 1.1 und alle laufzeitspezifischen Dialoge sind für die folgenden Sprachen lokalisiert:

- Traditionelles Chinesisch
- Vereinfachtes Chinesisch
- Englisch
- Französisch

- Deutsch
- Italienisch
- Japanisch
- Koreanisch
- Brasilianisches Portugiesisch
- Russisch
- Spanisch

Funktionen von AIR 1.1

Änderungen am Application Descriptor

Auch wenn es nicht unbedingt nötig ist, wird empfohlen, den Application Descriptor für die Verbindung mit AIR 1.1 Runtime zu aktualisieren.

Ändern Sie dazu das Attribut `xmlns` in:

```
http://ns.adobe.com/air/application/1.1
```

Bitte beachten Sie, dass diese Änderung erforderlich ist, wenn Sie die neuen lokalisierten Werte im Descriptor oder neue Funktionen in AIR 1.1 (unten beschrieben) nutzen möchten.

AIR 1.1 unterstützt die Lokalisierung von zwei Elementen in der Application Descriptor-Datei:

- `name`
- `description`

Die Application Descriptor-Datei sollte UTF-8-kodiert sein, damit mehrere Sprachen dargestellt werden können.

Entwickler, die ihre Anwendungen nicht lokalisieren möchten, können den Namen und die Beschreibung als einfache Text-Child-Nodes der Elemente `name` und `description` einbeziehen, genauso wie AIR 1.0.-Entwickler, die ihre Anwendungen lokalisieren möchten, Child-Elemente `text` in den Elementen `name` und `description` verwenden können. Die Verwendung von beiden ist nicht erlaubt.

Beispiel für die Verwendung von `<text>`-Child-Elementen:

```
<name>
  <text xml:lang="en">Name der englischen Anwendung</text>
  <text xml:lang="fr">Name der französischen Anwendung</text>
  <text xml:lang="de">Name der deutschen Anwendung</text>
  <text xml:lang="de">Name der japanischen Anwendung</text>
  <text xml:lang="es">Name der spanischen Anwendung</text>
  <text xml:lang="pt">Name der portugiesischen Anwendung</text>
  <text xml:lang="it">Name der italienischen Anwendung</text>
  <text xml:lang="zh_CN">Name der Anwendung in vereinfachtem
Chinesisch</text>
  <text xml:lang="zh_TW">Name der Anwendung in traditionellem
Chinesisch</text>
  <text xml:lang="ko">Name der koreanischen Anwendung</text>
  <text xml:lang="ru">Name der russischen Anwendung</text>
</name>
```

API-Ergänzungen in AIR 1.1

SQL

- `SQLException.detailID` : int
- `SQLException.detailArguments` : Array

Der Wert der Eigenschaft `SQLException.details` wird in AIR nicht lokalisiert. Bei einigen Datenbank Anwendungen wie einem SQL DB-Administrationstool wäre es hilfreich, wenn diese Informationen lokalisiert wären. Die zusätzlichen `SQLException`-Eigenschaften ermöglichen es Entwicklern, diese Zeichenfolgen zu lokalisieren.

Beispiel:

```
var conn:SQLConnection = new SQLConnection();
var dbFile:File =
File.applicationStorageDirectory.resolvePath("DBSample.db");
// Datenbank hat eine Tabelle "Mitarbeiter" mit Spalten
// employeeID, firstName, lastName, birthday

conn.open();

try {
    var selectStatement:SQLStatement = new SQLStatement();
    selectStatement.sqlConnection = conn;
    selectStatement.text = "SELECT name FROM employee;";
    selectStatement.execute();
} catch (err:SQLException) {
    // da es keine Spalte "Name" gibt, wird ein Fehler ausgegeben
    localizeError(SQLException);
}

function localizeError(e:SQLException):void {
    var argsLength:int = e.detailArguments.length;
    switch (e.detailID) {
        case 2030:
            // Standarddetailzeichenfolge: "Trigger '%s' ist bereits
            vorhanden"
            // etwas tun
            break;
        // ... weitere Fälle ...
        case 2036:
            // Standarddetailzeichenfolge: "keine Spalte:
            '%s[.%s[.%s]]'"
    }
}
```

```

var colPath:String = "";
if (argsLength == 1) {
    colPath = e.detailArguments[0];
} else if (argsLength == 2) {
    colPath = e.detailArguments[0]+"."+e.detailArguments[1];
} else if (argsLength == 3) {
    colPath =
e.detailArguments[0]+"."+e.detailArguments[1]+"."+e.detailArguments[2];
}
// oder verwenden Sie die Regionsinformationen, um eine
lokalisierte Zeichenfolge zu erzeugen
displayLocalizedDetail("Spalte '" + colPath + "' existiert
nicht.");
break;
default:
    displayLocalizedDetail(e.details);
}
}

function displayLocalizedDetail(str:String):void {
    // Fehlerdetail anzeigen
}

```

NativeWindow

- `NativeWindow.supportsTransparency` : Boolean

Die Eigenschaft gibt an, ob das Betriebssystem ein transparentes Fenster zeichnen kann. Der Wert dieser Eigenschaft ist bei Mac OS und Windows immer "true". Der Wert dieser Eigenschaft bei Linux hängt von der Konfiguration der Linux-Distribution ab.

Capabilities

- `Capabilities.languages` : Array

Array von Sprachcodes, die der Benutzer über Systemsteuerungen oder Systemvoreinstellungen sequenziert hat, von den im Betriebssystem bevorzugtesten zu den am wenigsten bevorzugten.

Datei

- `File.spaceAvailable` : Number

Gibt den für die Verwendung in diesem Dateiverzeichnis verfügbaren Platz in Byte an. Falls das Dateiverzeichnis nicht existiert, wird 0 zurückgegeben. Wenn das Dateiojekt ungültig ist, wird eine Ausnahme `IllegalOperationError` ausgegeben.

Zertifikatmigration

Die im AIR 1.1 SDK enthaltene Version von ADT hat eine neue Option, die die Aktualisierung einer Anwendung von einem alten Zertifikat auf ein neues bei gleichzeitiger Beibehaltung der Identität der Anwendung ermöglicht (beispielsweise von einem selbst signierten Zertifikat zu einem verketteten Zertifikat):

```
> adt -migrate SIGNING-OPTIONS <air-file-in> <air-file-out>
```

Bitte beachten Sie, dass die Datei `<air-file-in>` zuerst mit dem alten Zertifikat signiert wird; das alte Zertifikat wird mithilfe des Befehls `-migrate` an die `SIGNING_OPTIONS` weitergegeben.

Dieser Befehl akzeptiert eine AIR-Datei als Eingabe und erstellt eine AIR-Datei als Ausgabe. Da eine AIR-Datei nur eine Migrationssignatur enthalten kann, liegt ein Fehler vor, wenn die ankommende Datei bereits eine enthält.

Installation von Adobe AIR

Installation der AIR 1.1 Runtime und der AIR-Anwendung

Für die Installation der AIR 1.1 Runtime und der AIR-Anwendung benötigen Sie Administratorrechte.

Nahtlose Installation

Die nahtlose Adobe AIR 1.1-Installationsfunktion erfordert Flash Player 9 Update 3 (Version 9.0.115). Für die nahtlose Installationsfunktion wird ein Sample-Badge bereitgestellt, das angepasst werden kann.

Ablaufdatum für die AIR Beta Runtime

Die Adobe AIR Alpha 1, Beta 1 und Beta 2 Runtimes sind abgelaufen. Wenn Sie Anwendungen ausführen möchten, die für diese abgelaufenen Runtimes erstellt wurden, müssen Sie die Anwendungen auf AIR 1.0.x oder 1.1 aktualisieren. Adobe AIR Beta 3 läuft am 1. November 2008 aus, und nach Ablauf müssen alle AIR-Anwendungen, die für die Beta 3 Runtime erstellt wurden, auf AIR 1.0.x oder 1.1 aktualisiert werden. Benutzer können Anwendungen, die für eine abgelaufene Runtime entwickelt wurden, nicht installieren oder ausführen.

Tipps und Tricks

Öffnen einer AIR-Anwendung mit Safari auf Mac OS

Wenn Sie bei Verwendung von Safari auf Mac OS X auf einen Link zur Installation einer AIR-Anwendung klicken, wird diese bei OS 10.4 (Tiger) auf den Desktop und bei Mac OS 10.5 (Leopard) in den Downloadordner heruntergeladen und nicht installiert. Sie können dann auf die heruntergeladene `.air`-Datei klicken, um die Anwendung zu installieren.

Einstellen des MIME-Typs auf Ihrem Webserver für AIR-Anwendungen

Damit Clientbrowser eine heruntergeladene AIR-Anwendung erkennen, muss der Webserver, auf dem sich die AIR-Anwendung befindet, den MIME-Inhaltstyp `application/vnd.adobe.air-application-installer-package+zip` der Dateierweiterung `".air"` zuweisen. Fügen Sie beispielsweise für einen Apache-Webserver Folgendes zum Abschnitt `AddType` hinzu:

```
AddType application/vnd.adobe.air-application-installer-package+zip .air
```

adl - Debugging-Support

Der Debugging-Support wird standardmäßig beim Starten einer Anwendung mithilfe von ADL aktiviert. Bei Verwendung des Debugging-Supports führt die Runtime zusätzliche Prüfungen durch und generiert zusätzliche Debugging-Informationen, beispielsweise Debugging-Informationen von Flex Builder, und zeigt auch die Dialoge für die nichtbehandelten Ausnahmen an. Außerdem kann die

Anwendung während der Verwendung des Debugging-Supports langsamer laufen. Die Verwendung von ADL mit dem Flag `-nodebug` deaktiviert den Debugging-Support. Die Ausführung des Flags `-nodebug` emuliert außerdem den Ausführungsmodus einer installierten Anwendung genauer.

Fehlerbehebung (für Endanwender)

PDF-Inhalte werden nicht angezeigt

Um PDF-Inhalte in AIR anzuzeigen, stellen Sie sicher, dass Adobe Reader 8.1 oder später installiert ist.

SQL-Datenbank und Mac OS 10.5 Time Machine-Funktion

Die Ausführung von Mac OS 10.5 Time Machine bei gleichzeitiger Ausführung bestimmter SQL-Datenbanktransaktionen kann zu Inkonsistenzen in Ihrer Datenbank führen. Der Workaround besteht darin, sicherzustellen, dass die Datenbank vom automatischen Backup durch Time Machine ausgeschlossen ist.

AIR-Anwendungen und MobyDock

MobyDock, eine Freeware-Anwendung für die Anpassung der Windows-Taskleiste, verhindert die Ausführung von AIR-Anwendungen. Schließen Sie MobyDock, während Sie AIR-Anwendungen installieren und ausführen.

Firefox wird nicht als Standardbrowser verwendet

Bei Windows Vista werden von der AIR-Anwendung geöffnete Webseiten (dies erfolgt durch Aufrufen der Methode `navigateToURL()`) möglicherweise nicht im Standardbrowser angezeigt, sondern es wird immer der Internet Explorer geöffnet. Ein Workaround für dieses Problem: Gehen Sie zu "Start" → "Standardprogramme" → "Programmzugriff festlegen", erweitern Sie "Benutzerdefiniert", markieren Sie die Optionsschaltfläche neben der Anwendung, die Sie als systemweiten Standardbrowser einstellen möchten (beispielsweise Mozilla Firefox) und übernehmen Sie die Einstellung.

Fehlerbehebung (für Entwickler)

HTTP-Beschränkungen bei Windows

Bei Windows gibt es keine Unterstützung für die automatische Umleitung von `PUT`-, `POST`- oder `DELETE`-Methoden. AIR unterstützt außerdem keine Dekomprimierung von Flat- oder gzip-Dateien von HTTP-Antworten.

Links zu Bildern werden nicht richtig geladen

Beim Klicken auf einen Link, bei dem ein Bild geladen wird (statt einer HTML-Seite), wird nicht das Bild, sondern die Bild-Bytes angezeigt.

Verwendung der Dateneigenschaft von NativeMenuItem

Für die Eigenschaft `NativeMenuItem.data` werden die Objekte `int` oder `Number` in dieser Version nicht unterstützt.

Nicht unterstützte URL-Schemata

Nicht unterstützte URL-Schemata schlagen beim Aufrufen von JavaScript ohne Meldung fehl. Wenn Sie Beispiel in der Security Sandbox der Anwendung auf einen Link zu `` klicken (der nicht unterstützt wird), wird keine Ausnahme ausgegeben; er schlägt ohne Meldung fehl.

Eigenschaft ContextMenu.visible

Die Einstellung der Eigenschaft `ContextMenu.visible` hat keine Auswirkung. Elemente werden immer angezeigt.

Unterstützung für wenig verbreitete Schriftzeichen für den China National Standard GB 18030-2000

Wenig verbreitete Schriftzeichen wie Tibetisch werden möglicherweise in der AIR-Anwendung nicht korrekt angezeigt.

Virtuelle Tastatur in SWF-Inhalt von HTML auf Tablet PC geladen

Die virtuelle Tastaturfunktion für einen Tablet PC funktioniert für eine AIR SWF- und HTML-Anwendung, aber nicht für SWF-Inhalte, die von einer HTML-Datei aus geladen werden.

Windows-64-Bit-Betriebssystem: Installation von AIR-Anwendungen im 64-Bit-Programmordner

Wenn eine AIR-Anwendung im Ordner "Programme" (64-Bit) installiert wird, wird ein Fehler ausgegeben, dass die Anwendung nicht installiert werden kann. Die Anwendung wird jedoch trotzdem im Ordner "Programme (x86)" installiert.

app: URL-Schema mit offenen PDF-Parametern auf dem Mac

Die Verwendung des `app:` URL-Schemas mit offenen PDF-Parametern (zum Beispiel `app:/myfiles/test.pdf#page=2`) beendet das Laden bei Mac OS nicht.

htmlText-Links auf Vista

Unter Windows Vista werden `htmlText`-Links in Flex `<mx:text>`-Elementen nur im Internet Explorer geöffnet, auch wenn Firefox als Standardbrowser eingestellt ist.

JavaScript-Datumsobjekte, die eine Sandbox Bridge passieren, verlieren ihren ursprünglichen Typ

Der Workaround besteht darin, die Daten zu serialisieren und Datum/Zeit als Millisekunden zu übergeben.

Die Einstellung der Breite oder Höhe eines iframe auf 100 % kann unerwartete Ergebnisse verursachen

Der Workaround besteht darin, `iframe` auf einen Wert unter 100 % zu setzen.

Die Spaltenaffinitäten DATE und DATETIME sind mit AIR Beta-Versionen nicht kompatibel

Die Spaltenaffinitäten "DATE" und "DATETIME" in AIR 1.0 oder später für SQL-Datenbanken haben sich geändert und sind mit Betaversionen nicht kompatibel. Wenn Sie vorhandene Datenbankdateien haben, die "DATE" und "DATETIME" als Affinität (Spaltentyp) verwenden, müssen diese Tabellen neu gefüllt werden, damit sie funktionieren.

```
// Tabelle wurde mit dem folgenden Post SQL – CREATE TABLE post erstellt
(id INTEGER PRIMARY KEY, post_date DATE, title TEXT, content TEXT,
author_id INTEGER);
```

```
// Daten aus der vorhandenen Tabelle auswählen und das Datum als
Ganzzahl formatieren (dies vermeidet eine falsche Umwandlung des
julianischen Kalenderdatums)
```

```
sql.text = "select id, cast (post_date as integer) as p_date from
post;";
```



```

sql.clearParameters();
sql.execute();
// Tabelle mit den neuen Werten aktualisieren
var result:SQLResult = sql.getResult();
feeds.dataProvider = result.data;
sql.text = "update post set post_date = :d where id = :id;";
for (i=0; i<result.data.length; i++)
{
    sql.parameters[":id"] = i;
    sql.parameters[":d"] = new Date(result.data[i].p_date);
    sql.execute();
}

```

Das Verhalten der Spaltenaffinität ist mit AIR Beta-Versionen nicht kompatibel

Wenn eine Spalte einen deklarierten Typ (Affinität) hat, kann die Spalte das erfolgreiche Einfügen einer Zeile verhindern. In AIR 1.0 und später werden Spaltentypen erzwungen. Wenn der für eine Spalte angegebene Typ nicht erfolgreich für die Deklaration des Spaltentyps (Affinität) konvertiert werden kann, schlägt die Aktualisierung/Einfügung fehl. Beispiel:

```

// Tabelle wurde wie folgt erstellt: SQL- CREATE TABLE x (a INTEGER
PRIMARY KEY, b INTEGER);
sql.text = "INSERT INTO x VALUES (1, 5.5);"
sql.execute(); // da der für Spalte b (5.5) angegebene Wert keine
Ganzzahl ist, schlägt das Einfügen fehl.
// Tabelle wurde wie folgt erstellt: SQL- CREATE TABLE y (a INTEGER
PRIMARY KEY, b DATE);
sql.text = "INSERT INTO y (1, '');"
sql.execute(); // da der für Spalte b (") angegebene Wert nicht in ein
Datum umgewandelt werden kann, schlägt das Einfügen fehl

```

Dock- und Menüleiste werden im Vollbildmodus bei der Überlagerung von Fenstern in Mac OS nicht ausgeblendet

Wenn ein überlagerndes Fenster über einem Vollbildfenster liegt, werden bei Mac OS die Anwendungsmenüleiste und das Systemdock nicht ausgeblendet. Um dies zu beheben, setzen Sie den `type` des verschiebbaren Fensters auf `NativeWindow.lightweight`.

NativeWindow.height wird im Resize-Ereignis falsch gemeldet

Die Eigenschaft `NativeWindow.height` wird beim Lesen in einem `Resize`-Ereignis-Handler falsch gemeldet. Der Workaround besteht darin, stattdessen die Eigenschaft `afterBounds.height` für das Ereignis zu verwenden.

Festlegen von idleThreshold

Wenn Sie die Eigenschaft `idleThreshold` des Objekts `NativeApplication` nach Registrierung für das Ereignis `userIdle` festlegen, erhalten Sie die Standardeinstellung `idleThreshold` statt dem eingestellten Wert. Der Workaround besteht darin, den Ereignis-Handler aus der Registrierung zu entfernen, den Schwellenwert einzustellen und ihn wieder in die Registrierung zu schreiben.

Tracing zur Konsole

Wenn es auf Ihrem System eine Datei `mm.cfg` mit dem Parameter `TraceOutputFileEnable=1` gibt, wird keine Trace-Ausgabe zur Konsole gesendet. Der Workaround besteht darin, diesen Parameter aus der Datei `mm.cfg` zu entfernen. Die Datei `mm.cfg` befindet sich in "c:\Dokumente und Einstellungen\

Weitergabe des Headers "Cookie" an `URLRequest.requestHeaders`

Die Anforderung zur Weitergabe des Headers "Cookie" an die Eigenschaft `requestHeaders` eines `URLRequest`-Objekts, wird nicht an den Server gesendet, wenn `manageCookies=true` ist.

Vollständiger Zertifikatpfad für die Code-Signierung erforderlich

Einige Zertifikatanbieter (beispielsweise VeriSign) bieten standardmäßig keinen vollständigen Zertifikatpfad. Dieser ist jedoch für die Signierung von AIR-Anwendungen erforderlich. Informationen zur Erstellung eines Keystore mit einer vollständigen Zertifikatkette finden Sie unter <http://access1.sun.com/techarticles/Keytool.html>.

Von der Methode `NativeWindow.globalToScreen()` zurückgegebenes Punktobjekt kann auf Sekundärbildschirmen einen falschen Y-Wert haben

Der von der Methode `NativeWindow.globalToScreen()` zurückgegebene Y-Wert ist möglicherweise nicht genau, wenn der Parameter bei bestimmten Konfigurationen mit mehreren Monitoren einen Punkt nicht widerspiegelt, der sich auf der Hauptbildschirmanzeige befindet.

`setAsDefaultApplication()` und Dateierweiterungsverknüpfung unter Windows

Wenn in Windows ein Benutzer eine Verknüpfungsanwendung für eine Erweiterung mithilfe des Explorers auswählt, schlägt bei Verwendung von

`NativeApplication.setAsDefaultApplication()` die Festlegung der AIR-Anwendung als Standardanwendung fehl.

`removeAsDefaultApplication()` und Dateierweiterungsverknüpfung unter Windows Vista und Mac

Wenn es keinen anderen Handler für die im System registrierte Dateierweiterung gibt, wird bei Verwendung von `NativeApplication.removeAsDefaultApplication()` die Dateierweiterungsverknüpfung bei der AIR-Anwendung nicht entfernt.

ADT-Verwendung des Proxys mit Authentifizierung

ADT funktioniert derzeit nicht mit Proxyeinstellungen, die eine Authentifizierung erfordern.

Registrierung des Dock-Symbolereignisses

Auf dem Mac führt die Verwendung von `NativeMenu.addItemAt(item, 0)` beim Docksymbol dazu, dass alle darunter liegenden Menüpunkte ihre Ereignis-Handler `Select` verlieren. Der Workaround besteht in der Verwendung von `NativeMenu.addItem()`. Damit der letzte Punkt als oberster Menüpunkt angezeigt wird, löschen Sie das alte Menü und erstellen anschließend ein neues Menü.

Breite und Höhe für `NativeWindow`

Beim Festlegen der Eigenschaften `Width`, `Height` oder `Bounds` eines `NativeWindow`-Objekts in einer synchronen Funktion ist der für `Height` gemeldete Wert falsch, wenn er im Handler eines `Resize`-Ereignisses markiert wird.

Aktualisieren von Namen/Werte-Paaren im verschlüsselten lokalen Speicher

Das verschlüsselte Namen/Werte-Paar des lokalen Speichers kann nicht aktualisiert werden, wenn der neue Wert ein Substring des ursprünglichen Werts ist. Wenn Sie beispielsweise versuchen, den

Wert "monkey" mit "mo" zu aktualisieren, wird der ursprüngliche Wert "monkey" zurückgegeben. Der Workaround besteht darin, den Wert zu löschen, bevor er mit einer neuen Zeichenfolge aktualisiert wird.