Ada Magica

Christoph Grein

Mit Ada durch dick und dünn

Neulich habe ich zufällig eine Unterhaltung zwischen Lady Ada und einem verzweifelten Kunden mitbekommen.

Adam: *Madam I'm Adam.*

Ada: *Adam, I'm Ada.* Nett Sie kennen zu lernen. Was kann ich für Sie tun?

Adam: Meine Gnädigste, ich hab da ein Problem, das mich noch ganz verrückt macht. Schauen Sie doch bitte mal die folgenden Vereinbarungen an:

```
type D is array (Integer range <>)
  of Boolean;
type A is access all D;

X : aliased D (1..3);
AX: A            := X'Access;
BX: A (X'Range)  := X'Access;
```

Ich kann einfach nicht begreifen, warum mein dussliger Übersetzer das nicht akzeptieren mag.
Dam', I'mad!

Ada: Nun beruhigen Sie sich doch, guter Freund… Reichen Sie mir Ihre Hand und ich werde Sie durch dick und dünn des RM führen.
Schauen Sie, Sie haben da einen uneingeschränkten Reihungstyp definiert und einen Zeiger auf seine Objekte. Da der Typ uneingeschränkt ist, müssen seine Objekte natürlich ihre Grenzen zusammen mit den Komponenten abspeichern. Und genau das muss auch der Zeiger tun. Das heißt, der Zeiger ist ein sogenannter fetter Zeiger, im Unterschied zu normalen, dünnen Zeigern, die nur auf die Reihung selbst zeigen, deren Grenzen aus anderen Gründen schon bekannt sind.
Ihr Objekt `X` ist allerdings beschränkt durch explizite Grenzen. Das `'Access`-Attribut liefert daher einen dünnen Zeiger. `AX` ist jedoch ein fetter Zeiger. Ebenso ist `BX` ein fetter Zeiger…
Können Sie mir noch folgen? Hallo? Sind Sie noch da?

Adam: *Am I, ma?* Ich bin mir da nicht so ganz sicher…

Ada: Nun, ich stelle doch im RM 3.10(10) ganz klar dar, dass der bezeichnete Subtyp von `A` wie der von `A(X'Range)` `D` ist, nicht wahr?

Adam: *…mim…* [unverständlich]

Ada: Mein lieber Adam, wir müssen noch weiter ausholen. Aus RM 3.3.1(8,9) folgt, dass der nominelle und der aktuelle Subtyp von `X` `D(1..3)` ist. Also ist der aktuelle Subtyp beschränkt.
`X` ist ein beschränktes Objekt, weil sein nomineller Subtyp beschränkt ist.
RM 3.10.2(27/1) verlangt, dass der bezeichnete Subtyp von A statisch mit dem nominellen Subtyp übereinstimmt.
RM 4.9.1(2) stellt fest, dass statisch passende Subtypen statisch passende Einschränkungen haben.

Adam: *I…*

Ada: Nein, unterbrechen Sie mich nicht. Wir sind fast da.
Für `AX` ist der erwartete Typ von `X'Access` `A`. Der bezeichnete Subtyp `D` von `A` passt nicht statisch zum nominellen Subtyp `D(1..3)` von `X`.
Für `BX` ist der erwartete Typ von `X'Access` ebenfalls `A`. Der bezeichnete Subtyp `D` von `A(X'Range)` passt nicht statisch zum nominellen Subtyp `D(1..3)` von `X`.

Adam: *…*

Ada: Da sind Sie aber sprachlos, gell?
Und jetzt kommt der letzte Schlag. Betrachten Sie mal das da:

```
Y : aliased D := X;
AY: A            := Y'Access;
BY: A (Y'Range)  := Y'Access;
```

Wenn ich Ihnen jetzt auf die Sprünge helfe und Ihnen sage, dass der nominelle Subtyp von `Y` `D` ist, sein aktueller aber `D(1..3)`, so dass `Y` ein durch seinen Anfangswert beschränktes Objekt ist, sollten Sie ganz allein herausfinden können, warum das jetzt richtig ist.
So, lieber Freund, jetzt werden Sie sich sicher fragen, wozu eine Einschränkung an einen Zeiger (wie in Ihrem Beispiel bei `A` durch `(X'Range)`) denn eigentlich gut sei. Nun, eigentlich zu nix – man findet solch ein Biest gewöhnlich nur in ACVC-Tests. Seine einzige Aufgabe ist, eine zusätzliche Bereichsprüfung zu erzwingen; der bezeichnete Subtyp wird dadurch nicht verändert.
… äh … Hallo? Wo sind Sie geblieben…?

Adam: [ist als Erleuchteter ins Nirwana eingegangen und hat sich in Luft aufgelöst]